INAUGURAL-DISSERTATION

zur
Erlangung der Doktorwürde
der
Gesamtfakultät für Mathematik, Ingenieur- und Naturwissenschaften
der
Ruprecht-Karls-Universität
Heidelberg

vorgelegt von

Lüth, Carsten, MSc.
aus Bonn

Tag der mündlichen Prüfung: _____

Translating Active Learning from Theory to Praxis for 3D Biomedical Segmentation

Betreuer: Prof. Klaus Maier-Hein

Abstract

Active Learning (AL) promises to reduce annotation costs by strategically selecting the most informative samples for labeling. However, a significant theory-practice gap has hindered its adoption in real-world applications, particularly in biomedical image segmentation where annotation costs are substantial. This thesis addresses this gap through rigorous evaluation methodology which also directly simulates practical use-case scenarios to obtain measurements of annotation savings in 3D medical imaging. This is done in several steps.

First, we formalize requirements for the evaluation of AL to ensure that the measurements allow for trustworthy insights with regard to the reduction in annotation effort that a practitioner can expect on his dataset, as well as a simplified decision framework based on the economic nature of AL.

Then, we identify and formalize critical evaluation pitfalls that have hindered the practical application of AL research and create a comprehensive framework for evaluating deep learning-based AL. Our analysis shows that AL enables additional efficiency gains on well-optimized pipelines, but the absolute improvements over random baselines are smaller than on suboptimally optimized pipelines. Further, we show that an adequate evaluation of the benefits of AL must evaluate it in combination with orthogonal techniques such as self-supervised learning and hyperparameter optimization.

Through systematic analysis of uncertainty estimation for semantic segmentation, we resolve key misconceptions in the field, demonstrating that ensemble methods provide superior epistemic uncertainty estimates that are in theory essential for AL, and clarifying that test-time augmentation models epistemic rather than aleatoric uncertainty as previously claimed. These insights directly inform query method design for segmentation tasks.

Building on these foundations, we develop nnActive, a comprehensive framework for AL deployment in 3D biomedical segmentation that integrates best practices while introducing domain-specific adaptations including partial annotation strategies and improved random baselines. Through nnActive, we document previously unreported phenomena and find insufficient evidence supporting current uncertainty-based AL methods over improved random strategies.

Finally, we develop ClaSP PE, an uncertainty-based query method combining class-stratified selection with scheduled power-noise injection, specifically designed to address common failure modes in biomedical segmentation. Critically, we validate ClaSP PE through roll-out evaluation on four held-out datasets across diverse anatomical structures and imaging modalities, explicitly simulating real-world deployment. This provides the strongest empirical evidence to date for practical AL effectiveness in 3D biomedical imaging, demonstrating consistent annotation reductions when AL is properly evaluated and carefully deployed.

This thesis establishes that AL, while not a universal solution, is likely to deliver measurable annotation savings as a final optimization step in well-engineered pipelines, providing the evaluation principles, methodological insights, and practical tools necessary for evidence-based AL deployment in biomedical image segmentation.

Kurzfassung

Aktives Lernen (Active Learning, AL) verspricht eine Reduzierung der Annotationskosten durch die strategische Auswahl der informativsten Samples für die Beschriftung. Allerdings hat eine erhebliche Kluft zwischen Theorie und Praxis die Einführung in realen Anwendungen behindert, insbesondere in der biomedizinischen Bildsegmentierung, wo die Annotationskosten erheblich sind. Diese Arbeit befasst sich mit dieser Kluft durch eine strenge Bewertungsmethodik, die auch praktische Anwendungsszenarien direkt simuliert, um Messungen der Annotationsersparnisse in der medizinischen 3D-Bildgebung zu erhalten. Dies erfolgt in mehreren Schritten.

Zunächst formalisieren wir die Anforderungen für die Bewertung von AL, um sicherzustellen, dass die Messungen zuverlässige Erkenntnisse hinsichtlich der Reduzierung des Annotationsaufwands liefern, den ein Praktiker für seinen Datensatz erwarten kann, sowie einen vereinfachten Entscheidungsrahmen auf der Grundlage der wirtschaftlichen Natur von AL.

Anschließend identifizieren und formalisieren wir kritische Bewertungsfallen, die die praktische Anwendung der AL-Forschung behindert haben, und schaffen einen umfassenden Rahmen für die Bewertung von Deep-Learning-basiertem AL. Unsere Analyse zeigt, dass AL zusätzliche Effizienzsteigerungen bei gut optimierten Pipelines ermöglicht, aber die absoluten Verbesserungen gegenüber zufälligen Baselines sind geringer als bei suboptimal optimierten Pipelines. Darüber hinaus zeigen wir, dass eine angemessene Bewertung der Vorteile von AL in Kombination mit orthogonalen Techniken wie selbstüberwachtem Lernen und Hyperparameteroptimierung erfolgen muss.

Durch eine systematische Analyse der Unsicherheitsschätzung für die semantische Segmentierung klären wir wichtige Missverständnisse in diesem Bereich auf und zeigen, dass Ensemble-Methoden überlegene epistemische Unsicherheitsschätzungen liefern, die theoretisch für AL unerlässlich sind, und stellen klar, dass Testzeit-Augmentationsmodelle epistemische und nicht, wie zuvor behauptet, aleatorische Unsicherheit liefern. Diese Erkenntnisse fließen direkt in die Gestaltung von AL Methoden für Segmentierungsaufgaben ein.

Auf dieser Grundlage entwickeln wir nnActive, ein umfassendes Framework für den Einsatz und die Evaluierung von AL in der 3D-biomedizinischen Segmentierung, das bewährte Verfahren integriert und gleichzeitig domänenspezifische Anpassungen wie partielle Annotationsstrategien und verbesserte zufällige Baselines einführt. Mit nnActive dokumentieren wir bisher nicht berichtete Phänomene und finden keine ausreichenden Belege für die Überlegenheit aktueller, auf Unsicherheit basierender AL-Methoden gegenüber verbesserten Zufallsstrategien.

Schließlich entwickeln wir ClaSP PE, eine auf Unsicherheit basierende Abfragemethode, die eine klassenweise Auswahl mit einer geplanten Power-Noise-Injektion kombiniert und speziell für die Behebung häufiger Fehlermodi in der biomedizinischen Segmentierung entwickelt wurde. Entscheidend ist, dass wir ClaSP PE durch eine Rollout-Bewertung anhand von vier zurückgehaltenen Datensätzen aus verschiedenen anatomischen Strukturen und Bildgebungsverfahren validieren, wobei wir den realen Einsatz explizit simulieren. Dies liefert den bislang stärksten empirischen Beweis für die praktische Wirksamkeit von AL in der 3D-biomedizinischen Bildgebung und zeigt eine konsistente Reduzierung der benoetigten Annotationen fuer das Training, wenn AL richtig bewertet und sorgfältig eingesetzt wird.

Diese Arbeit belegt, dass AL zwar keine universelle Lösung ist, aber als letzter Optimierungsschritt in gut konzipierten Pipelines wahrscheinlich zu messbaren Einsparungen bei den Annotationen führt, und liefert die Bewertungsgrundsätze, methodischen Erkenntnisse und praktischen Werkzeuge, die für einen evidenzbasierten Einsatz von AL in der biomedizinischen Bildsegmentierung erforderlich sind.

Acknowledgements

There are simply too many people to whom I owe gratitude to list them all here. As the African saying goes, *"It takes a village to raise a child"*—and I believe this extends naturally to *"It takes an entire community to raise a PhD."* Naturally, I have been both a child and a PhD-student, so I had the benefit of both the village and the research-community.

As in research, we stand on the shoulders of giants, building upon the foundations laid by those who came before us. Keeping this in mind, I wish to thank all of the researchers upon whose absolutely phenomenal work I could base my own work on.

In what follows, I endeavor to thank those who have been central to this work, categorized by the distinct roles they played throughout my PhD.

First and foremost, I wish to express my gratitude towards my supervisors, Paul Jaeger and Klaus Maier-Hein, for their guidance, encouragement, and for providing the infrastructure necessary for my research. Also for believing in me and giving me the opportunity to pursue this PhD. Their efforts sharpened my thinking and made me a better and self-reliant researcher.

I would also like to give a special thanks to Fabian Isensee for staying with the ALEGRA and nnActive projects, which occupied me during the majority of my PhD from beginning to end, and Jeremias Traub for joining forces with me and his relentless effort to really close the deal on the nnActive project. Further, I would also like to thank Till Bungert for supporting me with his superior coding aura, Kim-Celine Kahl for her quick wit and reliability, as well as Lukas Klein for helping me make every figure a work of art. Additionally, I would like to thank Lars Kraemer for his incredible patience while taking over the communication with our collaborators during the finalization of the ALEGRA project.

Again, I would like to thank Kim-Celine Kahl for all the learning during the time I supervised her master's thesis.

Then I would like to extend my thanks towards "my room F.03.018" with Lars Kraemer, Karol Gotkowski, Sebastian Zieger, Till Bungert, and Lukas Klein for making up with my shenanigans and me having to make up with their shenanigans. I will just leave a few points to keep the memories fresh: 'Schattenboxen', 'Liegestuetze', 'Apache207', 'Mettbroetchen', 'Mario Kart Evenings', 'heidelberg.ai Events', 'Guessing Games', 'HI Pipeline', 'Esperanza Madness'.

Special thanks also go to Silvia Dias Almeida, working together towards redefining COPD detection as anomaly detection, while being named the 'MIC seasonal fragrance commissioner' was a blast.

Finally, within the DKFZ there are just many people to whom I would like to express my thanks for all kinds of fun, nice time and encouragement (there are simply too many, but here I will list the most crucial troup with no particular order): David Zimmerer, Tassilo Wald, Tim Raedsch, Balint Kovacs, Michael Baumgartner, Max Rokuss, Benjamin Hamm, Jessica Kaechele, Santhosh Parampottupadam, Selen Erkan, Dasha Trofimova, Jens Petersen, Saikat Roy...

Outside of work, I would like to thank my best friend, Alexander Eckerlin, for his endless patience, fierce loyalty, as well as refreshing gym, bouldering, and 'Siedler von Catan' sessions (the latter of these were somehow always mysteriously won by his wife Jeanette).

Then I would like to thank my girlfriend Ebba, who never got tired of reminding me that there is a life outside of work and who supported me in every way possible.

Also, I wish to express my thanks to my parents Maren Schoch and Tobias Lüth for supporting me all the way throughout life. Especially, I would like to thank my mother for just *always being there* and simply taking the time.

Finally I, wish to express my thanks to my entire family for always being a motivating and supporting factor in my life in many ways.

Especially I wish to thank my grandparents:

Gero and Alke Lüth for the discussion at their dinner table about the difference between a hypothesis and a theory as well as their purely descriptive and predictive nature mixed in with the optimal way to make scrambled eggs and hearty laughter.

Helmut and Sigrid Schoch for being always incredibly clear in both their words and decisions

resulting in a calming atmosphere at home.

Grants

I gratefully acknowledge the computing time provided on the high-performance computer HoreKa by the National High-Performance Computing Center at KIT (NHR@KIT). This center is jointly supported by the Federal Ministry of Education and Research and the Ministry of Science, Research and the Arts of Baden-Württemberg, as part of the National High-Performance Computing (NHR) joint funding program (`https://www.nhr-verein.de/en/our-partners`). HoreKa is partly funded by the German Research Foundation (DFG).

Part of this work was funded by Helmholtz Imaging (HI), a platform of the Helmholtz Incubator on Information and Data Science, and by the Helmholtz Association Initiative and Networking Fund under the Helmholtz AI platform grant (ALEGRA (ZT-I-PF-5-121)).

Contents

Chapter 1

Introduction

> Hofstadter's Law: It always takes longer than you expect, even when you take into account Hofstadter's Law.
>
> Douglas Hofstadter

Medical image segmentation, which is the task of delineating anatomical structures such as organs, vessels, or tumors in medical scans, is fundamental to modern healthcare, enabling surgical planning, disease diagnosis, and treatment monitoring. These medical scans come in a multitude of modalities, including computed tomography (CT), magnetic resonance imaging (MRI) or microscopy. However, creating the training data necessary for state-of-the-art segmentation models presents a formidable challenge (Litjens et al., 2017): expert radiologists must manually trace structure boundaries slice-by-slice through three-dimensional volumes, a process that can require several hours per scan. The annotation of a single 3D volume based on the median income of a radiologist in the US[1] and 1 hour average time to segment a brain tumor (B. H. Menze et al., 2014) costs approximately 255 US-Dollar (H. Wang et al., 2024).

This annotation burden creates a critical bottleneck. Medical imaging datasets are consequently orders of magnitude smaller than their natural image counterparts. While datasets like ImageNet contain millions of labeled images (Russakovsky et al., 2015a), annotated medical imaging benchmarks often comprise only 50-200 scans (Antonelli et al., 2022). The scarcity stems not from lack of imaging data, as hospitals generate vast quantities of unlabeled scans, but from the prohibitive cost of expert annotation. With limited time of expert annotators and growing demand for specialized AI systems across diverse anatomical structures, imaging modalities, and pathologies, this annotation bottleneck fundamentally constrains the development and deployment of medical imaging AI. The challenge is compounded by domain-specific factors: inter-observer variability means different experts produce inconsistent annotations (Joskowicz et al., 2019), pathological cases require even more specialized expertise, and privacy regulations restrict data sharing, preventing the consolidation that has benefited natural image domains. These factors make annotation efficiency not merely desirable but essential for practical medical imaging AI deployment.

Active Learning (AL), is a machine learning paradigm where the model actively selects which samples from an unlabeled pool should be annotated, rather than relying on random or externally determined sampling (Settles, 2009). The core hypothesis is that carefully selected training examples can achieve comparable or superior model performance with substantially fewer labeled samples than random selection. By querying the most informative instances, specifically those where the model is uncertain or which cover underrepresented regions of the input space, AL aims to maximize the value extracted from each annotation. The motivation for AL is fundamentally economic and lies in cost savings during the annotation process. Importantly, AL is orthogonal to other annotation efficiency techniques. Transfer learning from pre-trained models (Radford, J. W. Kim, et al., 2021), self-supervised learning on unlabeled data (T. Chen et al., 2020), and semi-supervised learning

[1] https://www.salary.com/tools/salary-calculator/radiologist-hourly

(Sohn, Berthelot, C.-L. Li, et al., 2020) can all be combined with AL: these methods provide better model initialization and training procedures, while AL determines which samples to annotate. This orthogonality means that AL can provide additional efficiency gains on top of already optimized training pipelines, making it relevant even as foundation models and other techniques advance. Because machine learning fundamentally requires labeled examples that define how tasks should be solved, strategic data selection through AL when providing benefits upon application remains valuable regardless of architectural or training innovations.

Despite decades of research and hundreds of papers proposing query strategies (Settles, 2009; Gal, Islam, et al., 2017a; Kirsch, van Amersfoort, et al., 2019a), AL has seen limited adoption in real-world deep learning workflows (Settles, 2011; Tomanek and Olsson, 2009; Jaster and Kohlhase, 2025; Abraham and Dreyfus-Schmidt, 2021). For medical segmentation even leading literature reviews (Budd et al., 2021; H. Wang et al., 2024) do not give clear examples of real-world applications of AL or a clear recommendation for its use.

This gap stems from several factors, but most crucially practitioners face a fundamental methodological challenge: determining which query strategy works best for their specific problem would require labeling multiple AL trajectories, directly contradicting AL's purpose of reducing annotation effort. Unlike transfer learning or data augmentation, whose benefits can be validated using standard train-test splits, AL's value proposition cannot be verified without extensive labeling that defeats its purpose. This means that practitioners need to perform their own dedicated AL evaluations on similar problems or solely rely on the results of the literature. Nearly all AL research (Settles, 2009; Settles, 2011; H. Wang et al., 2024) employs simulation studies where true labels for entire datasets are known in advance, allowing researchers to simulate annotation by selectively revealing ground truth.

While this approach enables systematic comparison of query strategies, it raises critical questions: Do findings from these controlled experiments transfer to real-world deployment? Are there subtle differences between simulation and practice, such as implementation constraints, that affect AL's practical utility?

The gap between simulation conditions and deployment reality creates uncertainty for practitioners considering AL adoption. Further, AL performance is highly dependent on problem characteristics: dataset distribution, task difficulty, model architecture, annotation budget, and query size all influence which query strategies succeed (Munjal et al., 2022a; Mittal, J. Niemeijer, et al., 2023; Settles, 2009). A method demonstrating strong performance on natural image classification may fail on medical image segmentation, or vice versa. Without clear guidance on when and why different strategies work, practitioners cannot confidently predict whether AL will benefit their specific use case. This uncertainty, combined with AL's inherent implementation complexity and computational overhead, makes adoption risky (Romberg et al., 2025). If AL fails to deliver annotation savings in practice, its adoption leads to resource expenditure, including implementing the AL pipeline, multiple training rounds, and computational costs, with no benefit. Worse, poor query strategies could potentially harm performance by selecting misleading or redundant examples. Experienced practitioners are acutely aware of this risk and hesitate to adopt AL based solely on simulation results that may not generalize to their specific problem. These challenges highlight a central issue: the field lacks robust evaluation methodologies that can convince practitioners that AL will provide practical benefits for their specific domain and use case (Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B. Chan, 2022b). Addressing this theory-practice gap requires not just better query strategies, but better frameworks for evaluating, understanding, and predicting when AL will succeed.

This thesis addresses the theory-practice gap by developing rigorous evaluation methodologies for Active Learning and providing practitioners with evidence-based guidance for AL adoption. We focus on both image classification and 3D biomedical segmentation, as these tasks represent complementary challenges: classification provides a well-understood baseline for studying AL fundamentals, while 3D biomedical segmentation represents a domain where AL is critically needed due to substantial annotation costs in medical images.

1.1 Research Questions & Outline

To bridge the gap between AL theory and practical deployment, this thesis addresses the following research questions:

RQ1 What are general aspects necessary for the evaluation of Active Learning methods?

- Discussed in chapter 4 by reasoning from first principles to obtain requirements extending across domains and tasks.

RQ2 How can we ensure generalizable and meaningful evaluation of Active Learning methods to guide practitioners in deep active classification?

- In chapter 5, we reveal pitfalls of Active Learning evaluation and build a framework for meaningful evaluation.

RQ3 How can systematic analysis of uncertainty estimation for semantic segmentation inform the design of Active Learning query methods?

- In chapter 6, we reveal pitfalls of the evaluation of uncertainty estimation and build a framework for meaningful evaluation revealing.

RQ4 How can we evaluate Active Learning in 3D biomedical segmentation to ensure that measurements of annotation effort reductions are both realistic and transferable to new applications?

- In chapter 7, we reveal pitfalls of the evaluation of Active Learning and build nnActive a framework and benchmark for meaningful evaluation.

RQ5 Can we develop an uncertainty-based Active Learning method that reduces annotation effort in Active Learning for 3D biomedical segmentation while generalizing to novel datasets?

- In chapter 8, we build upon the learnings from nnActive and propose a novel query method.

These questions progress from establishing evaluation principles (**RQ1**) through rigorous methodology development for classification (**RQ2**) and segmentation-specific considerations (**RQ3**), to practical validation on biomedical data (**RQ4-RQ5**). Together, they provide a systematic framework for understanding when, why, and how AL can deliver practical benefits in real-world deployment scenarios.

1.2 List of Publications

1.2.1 Publications Relevant to this Thesis

The following publications were published that are relevant to this thesis:

P1. Carsten Tim Lüth, Till J. Bungert, Lukas Klein, and Paul F Jaeger (2023). "Navigating the Pitfalls of Active Learning Evaluation: A Systematic Framework for Meaningful Performance Assessment". In: *Thirty-Seventh Conference on Neural Information Processing Systems*

P2. Kim-Celine Kahl, Carsten T. Lüth, Maximilian Zenk, Klaus Maier-Hein, and Paul F Jaeger (2024a). "ValUES: A Framework for Systematic Validation of Uncertainty Estimation in Semantic Segmentation". In: *The Twelfth International Conference on Learning Representations*
(shared first-authorship, Oral Presentation – top 1%)

P3. Carsten T. Lüth, Jeremias Traub, Kim-Celine Kahl, Till J. Bungert, Lukas Klein, Lars Krämer, Paul F. Jaeger, Fabian Isensee, and Klaus Maier-Hein (2025). "nnActive: A Framework for Evaluation of Active Learning in 3D Biomedical Segmentation". In: *Transactions on Machine Learning Research*
(shared first-authorship)

P4. Carsten T. Lüth, Jeremias Traub, Kim-Celine Kahl, Till J. Bungert, Lukas Klein, Lars Krämer, Paul F. Jaeger, Klaus Maier-Hein, and Fabian Isensee (2025). "Finally outshining the Random Baseline: A simple and effective solution for Active Learning in 3D biomedical imaging". In: *Submitted to Transactions on Machine Learning Research*. Under review
(shared first-authorship)

1.2.2 Further Publications

The following publications were published during this PhD which are not directly related to this thesis spanning a variety of topics in the field of Machine Learning, Computer Vision and Medical Imaging.

- **Anomaly Detection for Biomedical Imaging**
  1. Carsten T. Lüth, David Zimmerer, Gregor Koehler, Paul F. Jaeger, Fabian Isensee, Jens Petersen, and Klaus H. Maier-Hein (Jan. 2023). *CRADL: Contrastive Representations for Unsupervised Anomaly Detection and Localization*. arXiv: `2301.02126 [cs]`
  2. Silvia D Almeida, Carsten T Lüth, Tobias Norajitra, Tassilo Wald, Marco Nolden, Paul F Jäger, Claus P Heussel, Jürgen Biederer, Oliver Weinheimer, and Klaus H Maier-Hein (2023). "cOOpD: Reformulating COPD Classification on Chest CT Scans as Anomaly Detection Using Contrastive Representations". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 33–43
  (shared first-authorship, MICCAI STAR Award)
  3. Silvia D Almeida, Tobias Norajitra, Carsten T Lüth, Tassilo Wald, Vivienn Weru, Marco Nolden, Paul F Jäger, Oyunbileg von Stackelberg, Claus Peter Heußel, Oliver Weinheimer, et al. (2024). "Prediction of disease severity in COPD: a deep learning approach for anomaly-based quantitative assessment of chest CT". in: *European radiology* 34.7, pp. 4379–4392
  4. Silvia D. Almeida, Tobias Norajitra, Carsten T Lüth, Tassilo Wald, Vivienn Weru, Marco Nolden, Paul F Jäger, Oyunbileg von Stackelberg, Claus Peter Heußel, Oliver Weinheimer, et al. (2024). "How do deep-learning models generalize across populations? Cross-ethnicity generalization of COPD detection". In: *Insights into imaging* 15.1, p. 198
- **Failure Detection**
  1. Paul F Jaeger, Carsten Tim Lüth, Lukas Klein, and Till J. Bungert (2023). "A Call to Reflect on Evaluation Practices for Failure Detection in Image Classification". In: *The Eleventh International Conference on Learning Representations*
  (Oral Presentation – top 1%)
  2. Jeremias Traub, Till J. Bungert, Carsten T. Lüth, Michael Baumgartner, Klaus H. Maier-Hein, Lena Maier-Hein, and Paul F. Jäger (2024). "Overcoming Common Flaws in the Evaluation of Selective Classification Systems". In: *Advances in Neural Information Processing Systems*. Ed. by A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang. Vol. 37. Curran Associates, Inc., pp. 2323–2347

(Spotlight – top 5%)

- **Vision-Language Models (VLMs)**
  1. Kenza Amara, Lukas Klein, Carsten Lüth, Paul Jäger, Hendrik Strobelt, and Mennatallah El-Assady (2024). "Why context matters in VQA and Reasoning: Semantic interventions for VLM input modalities". In: *arXiv preprint arXiv:2410.01690*
  2. Kim-Celine Kahl, Selen Erkan, Jeremias Traub, Carsten T. Lüth, Klaus Maier-Hein, Lena Maier-hein, and Paul F Jaeger (2025). "SURE-VQA: Systematic Understanding of Robustness Evaluation in Medical VQA Tasks". In: *Transactions on Machine Learning Research*
- **Explainable AI**
  1. Lukas Klein, Carsten Lüth, Udo Schlegel, Till Bungert, Mennatallah El-Assady, and Paul Jäger (2024). "Navigating the maze of explainable ai: A systematic approach to evaluating methods and metrics". In: *Advances in Neural Information Processing Systems* 37, pp. 67106–67146

Chapter 2

Background

> To consult the statistician after an experiment is finished is often merely to ask him to conduct a post mortem examination. He can perhaps say what the experiment died of.
>
> Ronald Fisher

This section provides the general brackground and notation required for this thesis.

2.1 Information Theory

Information Theory provides a mathematical framework for quantifying information, uncertainty, and dependence within data. Originally introduced by Shannon (1948), it forms the theoretical foundation for many concepts used in modern machine learning, including uncertainty estimation, model calibration, and probabilistic inference. In the context of AL and uncertainty-based decision making, Information Theory offers a principled way to formalize what it means for a model to be uncertain, how much information an observation provides, and how learning changes beliefs about unknown quantities.

In the following, we introduce the fundamental quantities and notation used throughout this thesis. We closely follow the unified notation proposed by Kirsch and Gal, 2021 to maintain consistency between probabilistic and information-theoretic expressions.

2.1.1 Notation and Basic Definitions

Notation: Throughout this thesis, we use the following notation:

- X, Y: Random variables representing data
- x, y: Specific outcomes of random variables X and Y
- θ: Parameter vector in $\mathbb{R}^d$
- $p(\cdot)$: Probability distribution
- $h(\cdot)$: Shannon's information content
- $H(\cdot)$: Shannon's entropy
- $H(\cdot||\cdot)$: Cross-entropy
- $D_{KL}(\cdot||\cdot)$: Kullback-Leibler divergence
- $I(\cdot;\cdot)$: Mutual Information

When the probability distribution p is understood from context, we simplify the notation to enhance readability:

- $H[X] := H(p(X))$
- $H[y] := h(p(y))$ for a specific outcome y

Fundamental Quantities: Given a random variable X with probability distribution $p(x)$, a non-negative function q, and a non-negative real number ρ, we define:

- Information Content: $h(\rho) := -\ln \rho$
- Cross-Entropy: $H(p(X)||q(X)) := \mathbb{E}_{p(x)}[h(q(x))]$
- Entropy: $H(p(X)) := H(p(X)||p(X))$

Interpretation: Entropy measures the expected information content or uncertainty inherent in a random variable (Shannon, 1948). The cross-entropy provides a bridge between these quantities, representing the average number of bits required to encode samples from p when using the distribution q for encoding.

2.1.2 Conditional Entropy and Outcomes

For random variables X and Y with an observed outcome y of Y, we define entropy measures over specific outcomes rather than distributions:

- Joint entropy with outcome: $H[X, y] := \mathbb{E}_{p(x|y)}[h(p(x, y))]$
- Conditional entropy given outcome: $H[X|y] := \mathbb{E}_{p(x|y)}[h(p(x|y))]$
- Conditional entropy of outcome: $H[y|X] := \mathbb{E}_{p(x)}[h(p(y|x))]$

These definitions allow us to reason about the information content of specific observed outcomes y rather than averaging over all possible realizations of Y, which is particularly useful in AL scenarios where we evaluate specific potential observations.

2.1.3 Mutual Information

Mutual Information (MI) quantifies the reduction in uncertainty about one random variable given knowledge of another. It measures the strength of statistical dependency between two variables and is fundamental to information-theoretic approaches in AL.

Definitions:

- Point-Wise Mutual Information: $I[x; y] := H[x] - H[x|y] = h\left(\frac{p(x)p(y)}{p(x, y)}\right)$
- Mutual Information: $I[X; Y] := H[X] - H[X|Y] = \mathbb{E}_{p(x,y)}[I[x; y]]$
- Information Gain: $I[X; y] := H[X] - H[X|y]$
- Surprise: $I[y; X] := H[y] - H[y|X]$

Interpretation: Mutual information captures, on average, how much observing one variable reduces uncertainty about another. All MI measures are positive semi-definite, with higher values indicating stronger predictive power. The non-commutative variants—information gain and surprise—are particularly relevant for model-based uncertainty quantification. Information gain measures how much a specific observation y reduces uncertainty about X, while surprise quantifies how unexpected the outcome y is given knowledge of X. These asymmetric measures are crucial for AL strategies that select observations to maximize information gain.

2.1.4 Fisher Information

The Fisher Information, proposed by Fisher (1922), is an information-theoretic quantity that describes the amount of information an observable random variable Y carries about an unknown parameter $\theta \in \mathbb{R}^d$ upon which the probability of Y depends.

Definition: The Fisher Information is defined as the variance of the score function $s(\theta, Y) = \nabla_\theta \log p(Y|\theta)$:

$$\mathcal{I}(\theta) = \mathbb{E}_{p(Y|\theta)} \left[(\nabla_\theta \log p(Y|\theta))(\nabla_\theta \log p(Y|\theta))^T \right] \tag{2.1}$$

The Fisher Information matrix $\mathcal{I}$ is always positive semi-definite, reflecting that information cannot be negative. Under mild regularity conditions, when the log-likelihood is twice differentiable, it can equivalently be expressed as the expected negative Hessian of the log-likelihood (Lehmann and Casella, 1998, eq. 2.5.16):

$$\mathcal{I}(\theta) = -\mathbb{E}_{p(Y|\theta)} \left[\nabla^2_\theta \log p(Y|\theta) \right] \tag{2.2}$$

Interpretation: For scalar parameters θ, the Fisher Information has an intuitive interpretation. A high value indicates that the score function fluctuates strongly, meaning small changes in θ lead to large changes in the likelihood. This makes the data highly informative about θ because the likelihood function is sharply peaked. Conversely, a small value indicates that the likelihood is relatively flat with respect to θ, so changes in the parameter have little effect on the model's predictions.

Application to Predictive Models: For a predictive model, Fisher Information can be interpreted as the expected observed information based on the model's own predictions $p(y|x, \theta)$ (Kirsch and Gal, 2022):

$$\mathcal{I}(\theta) = -\mathbb{E}_{p(y|x,\theta)} \left[\nabla^2_\theta \log p(y|x, \theta) \right] = H''[Y|x, \theta] \tag{2.3}$$

where $H''[y|x, \theta] = -\nabla^2_\theta H[y|x, \theta]$ denotes the Hessian of the entropy with respect to the parameters. This formulation connects Fisher Information to the curvature of the predictive distribution's entropy, providing a natural link to uncertainty quantification in AL.

2.2 Computer Vision

Computer vision aims to endow artificial agents with the ability to perceive, interpret, and act upon visual information from their environment (Torralba et al., 2024). Beyond practical applications in autonomous driving, robotics, or medical imaging, computer vision serves as a core research domain for investigating how visual representations can be learned from data, how uncertainty can be quantified, and how data efficiency can be improved through methods such as AL.

At its core, computer vision focuses on acquiring, processing, and analyzing sensory input–such as images, video sequences, multi-camera setups, or 3D scans–to extract geometric and semantic information that enables reasoning about the physical world. The field encompasses a hierarchy of tasks, from low-level image restoration and structure-from-motion to high-level reasoning problems such as object detection, recognition, and semantic segmentation. Formally, these problems can be described as learning a mapping from visual observations to structured scene represetations enabling the agent to interpret and upon its environment.

In this thesis, we focus on two central subfields of computer vision, namely object recognition, which also refer to interchangeably as classification, and semantic segmentation, both of which provide complementary views on visual understanding as we show in fig. 2.1.

2.2.1 Object Recognition

Object recognition which we will refer to interchangeably as classification refers to the task of assigning one or more semantic labels to the entities present in an image. In this thesis, we will only dicuss the case where each image has one corresponding label. It is among the most fundamental problems in computer vision and serves as a proxy for learning rich visual representations that can generalize across domains and downstream tasks.

The most widely used performance metric in object recognition is the accuracy indicated by its use in multiple standard image classification benchmarks (Russakovsky et al., 2015a; A. Krizhevsky, 2009). The accuracy quantifies the ratio of correctly classified samples to the total number of samples. More thoroughly defined, when C denotes the number of classes with TP_c being the

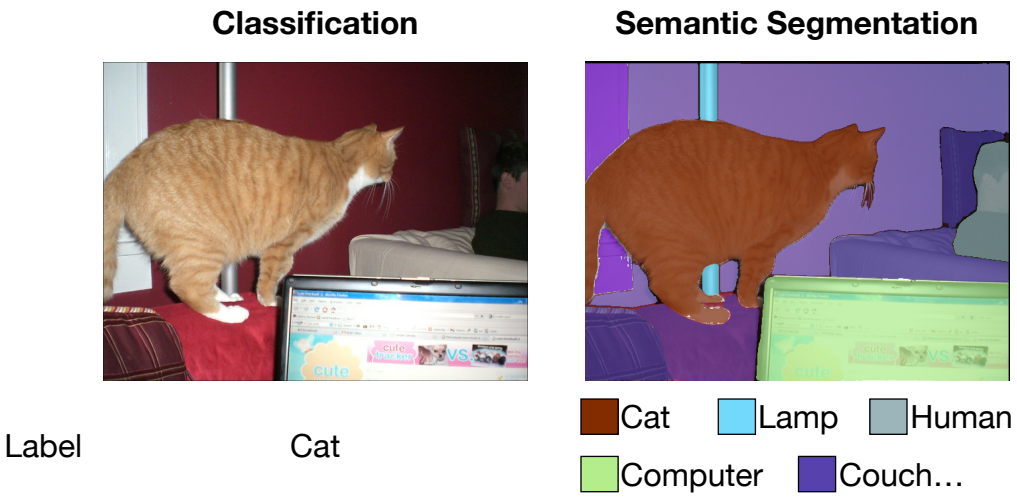


Figure 2.1: Classification and Semantic Segmentation explained by means of the same example image. Whereas in Classification the label is only the entity which occupies the image, in semantic segmentation the location of multiple different entities inside of the image is used as label.

number of true positives, FN_c being the number of false negatives and FP_c being the number of false positives belonging to class c over the entire dataset, it is defined as:

$$\text{Accuracy} = \frac{\sum_{c=1}^{C} \mathrm{TP}_c}{\sum_{c=1}^{C} (\mathrm{TP}_c + \mathrm{FN}_c)}. \tag{2.4}$$

While accuracy is well-suited for balanced datasets, it can be misleading in the presence of strong class imbalance, where rare classes contribute only marginally to the final score. This makes the accuracy a good choice for balanced datasets but reduces its suitability for imbalanced datasets, especially when the rare classes represent cases that need to be caught such as credit fraud in financial data or diseases in medical data. The balanced accuracy or mean recall (L. Maier-Hein et al., 2024) represents one metric which gives equal weight to all classes that is defined as:

$$\text{Mean Recall} = \sum_{c=1}^{C} \frac{1}{C} \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FN}_c}. \tag{2.5}$$

Another common metric for imbalanced scenarios is the macro averaged F1-score (L. Maier-Hein et al., 2024) which is the harmonic mean of precision and recall and defined as:

$$F_1 = \sum_{c=1}^{C} \frac{1}{C} \frac{2\mathrm{TP}_c}{2\mathrm{TP}_c + \mathrm{FP}_c + \mathrm{FN}_c}. \tag{2.6}$$

All of these performance metrics share that they are scalar values which range from 0 to 1 with larger values indicating better performance and are commonly computed over the entire dataset.

Historically, object recognition methods relied on engineered feature extractors such as Scale Invariant Feature Transform (Lowe, 2004) or histogramms of oriented gradients (Dalal and Triggs, 2005), followed by shallow classifiers like Support Vector Machines (Cortes and Vapnik, 1995). The paradigm shift occurred with the introduction of deep convolutional neural networks, particularly with AlexNet (Alex Krizhevsky et al., 2017), which demonstrated the power of hierarchical representation learning on large datasets such as ImageNet (Russakovsky et al., 2015a). Subsequent architectures, such as the Residual Neural Network (ResNet) (K. He, X. Zhang, et al., 2016) and the Vision Transformers (Dosovitskiy et al., 2021), further advanced both model capacity and generalization.

From a learning-theoretic perspective, object recognition represents a well-defined supervised learning setup where the objective is to minimize the empirical risk over a labeled dataset. As each training sample requires only a single categorical label, annotation costs are relatively low compared to more complex tasks such as segmentation or detection. This property makes object recognition

a preferred experimental setting for AL studies, as it isolates the effect of data selection strategies from high annotation variability. Consequently, many AL benchmarks (Mittal, Tatarchenko, et al., 2019a; Beck et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B Chan, 2022a; Gal, Islam, et al., 2017b; Kirsch, van Amersfoort, et al., 2019b) have been established on object recognition datasets to study query informativeness, model uncertainty, and sample diversity.

2.2.2 Semantic Segmentation

The goal of semantic segmentation is to obtain a dense segmentation mask which carries the semantic information which categories called classes are located inside of an image. In case of a 2D image the the segmentation describes to which class each pixel of the image belongs. In contrast to object recognition, which yields a single global label per image, semantic segmentation provides a spatially resolved understanding of the scene by determining what is present and where it occurs. This makes it a core problem in computer vision with critical applications in domains such as autonomous driving, medical imaging, and robotics.

Most commonly metrics evaluate for each image how much the predicted segmentation mask and the ground truth mask overlap. Gebnerally the mean over these metrics over the entire dataset is reported which differentiates this from object recognition. The most predominant metric for natural images is the mean Intersection over Union (mIoU) metric (Z. Wang et al., 2023) which is defined as follows:

$$\text{mIoU} = \frac{1}{C}\sum_{c=1}^{C}\frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c + \text{FN}_c}. \tag{2.7}$$

Alternatively, the mean Dice coefficient (or F1 score for segmentation) (Dice, 1945) is used, particularly in medical imaging, to quantify spatial similarity between predicted and annotated regions:

$$\text{Dice} = \frac{1}{C}\sum_{c=1}^{C}\frac{2\text{TP}_c}{2\text{TP}_c + \text{FP}_c + \text{FN}_c}. \tag{2.8}$$

These metrics emphasize the quality of spatial agreement rather than simple classification correctness, making them suitable for highly imbalanced or spatially complex datasets.

Historically, semantic segmentation approaches relied on hand-crafted features and probabilistic graphical models, such as Markov Random Fields and Conditional Random Fields (Shotton et al., 2006). A major breakthrough came with the introduction of Fully Convolutional Networks (J. Long et al., 2015), which enabled end-to-end dense prediction by replacing fully connected layers with convolutional upsampling. Subsequent architectures, including U-Net (Ronneberger et al., 2015), DeepLab (L.-C. Chen et al., 2017), , HR-Net (Jingdong Wang et al., 2020) and SegFormer (E. Xie et al., 2021), advanced the field by improving multi-scale feature aggregation, receptive field design, and computational efficiency. These innovations have made deep segmentation models the de facto standard for dense prediction tasks.

From a research perspective, semantic segmentation presents a considerably more challenging setup than object recognition. Unlike classification, where predictions are independent per image, the label space in segmentation is structured and spatially correlated, making the learning process sensitive to factors such as annotation noise, boundary precision, and class imbalance. Moreover, the computational complexity of segmentation models is typically higher, as they must predict dense label maps rather than single categorical outputs, requiring more memory and compute resources during both training and inference. Finally, the annotation cost per sample is substantially greater: the manual delineation of pixel-level masks by experts can take several orders of magnitude longer than assigning a single class label.

This combination of high annotation cost and structured output space makes semantic segmentation a particularly compelling yet challenging use case for AL (Mittal, Tatarchenko, et al., 2019a).

2.2.3 Semantic Segmentation for Biomedical Images

Biomedical image segmentation represents a specialized and critical application domain of semantic segmentation, where the goal is to delineate anatomical structures, tissues, or pathological regions

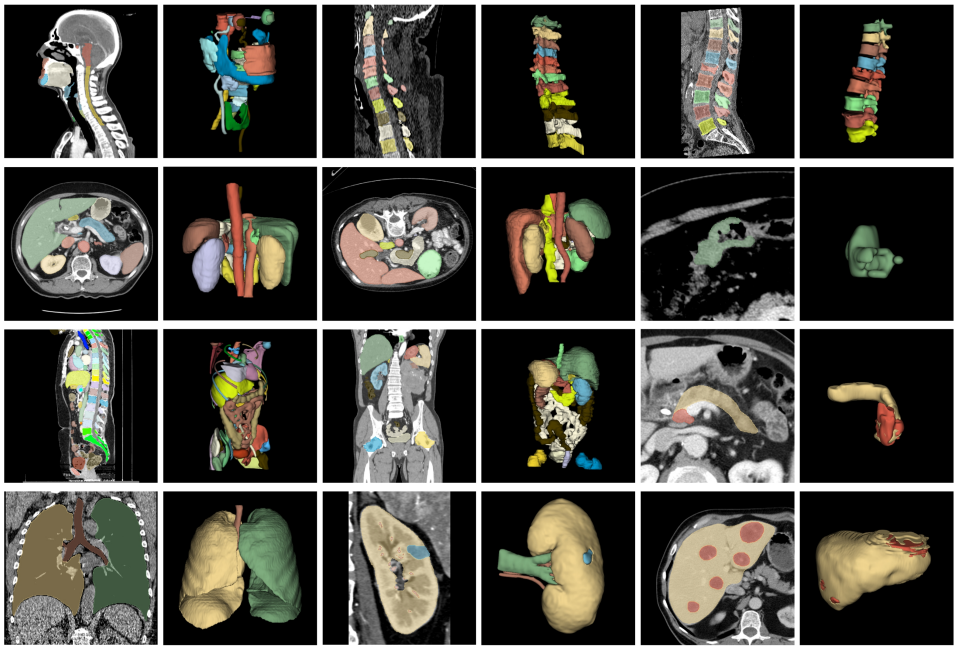


Figure 2.2: Volume examples of CT images containing various anatomical structures as 2D slices and 3D shapes in the images. Taken from Du et al. (2024).

in medical imaging data (Antonelli et al., 2022). While sharing the fundamental objective of dense pixel- or voxel-wise classification with natural image segmentation, biomedical segmentation presents unique challenges and characteristics that distinguish it from general-purpose computer vision tasks.

The primary distinction lies in the nature of the data: in this thesis, we will focus on biomedical images that are three-dimensional volumetric scans acquired through modalities such as Computed Tomography (CT), Magnetic Resonance Imaging (MRI), or microscopy techniques (Litjens et al., 2017). Unlike 2D natural images, these 3D volumes have fixed spatial resolutions and capture the relationships between anatomical structures, requiring segmentation methods to reason about volumetric context rather than planar information alone. A CT scan of the abdomen, for instance, may consist of hundreds of 2D slices that together form a 3D representation of organs, each requiring consistent segmentation across the entire volume.

Medical images exhibit fundamentally different visual characteristics than natural images. Tissue contrast, intensity distributions, and texture patterns are determined by physical imaging principles rather than natural lighting and surface properties. Moreover, different imaging modalities (CT, MRI, ultrasound) produce different appearances for the same anatomical structures.

Pathological regions or small anatomical structures often occupy only a tiny fraction of the total image volume. In tumor segmentation, for example, the target region may constitute less than 1% of the total voxels, making metrics like accuracy misleading and requiring specialized loss functions and evaluation metrics.

Unlike objects in natural images with clear edges, anatomical structures often have fuzzy or indistinct boundaries. Soft tissue transitions, partial volume effects, and image artifacts can make even expert annotations inconsistent, particularly at structure boundaries. Further, expert annotations exhibit notable variability, even among experienced radiologists (S. A. A. Kohl et al., 2019). Different experts may delineate structures differently, and the same expert may produce inconsistent annotations at different times, introducing label noise that complicates model training and evaluation.

The volumetric nature of biomedical data necessitates specialized architectural and methodological considerations:

2D models. The simplest strategy treats each 2D slice independently, applying standard 2D segmentation networks. While computationally efficient, this approach ignores inter-slice context and can produce inconsistent predictions across the volume depth.

2.5D models. These methods incorporate limited 3D context by processing neighboring slices jointly (e.g., treating adjacent slices as color channels) while maintaining 2D architectures. This provides a compromise between computational efficiency and volumetric reasoning.

3D models. 3D models process the entire volumetric context, enabling the model to learn spatial relationships across all three dimensions. The 3D U-Net (Çiçek et al., 2016) and its variants represent the dominant architecture family for this approach, extending the encoder-decoder structure to volumetric data. However, 3D convolutions are memory-intensive, often requiring patch-based processing or downsampling to fit within GPU constraints.

Due to extreme class imbalance and boundary ambiguity, biomedical segmentation relies heavily on metrics that are robust to these characteristics. The Dice coefficient (eq. (2.8)) is the most widely used metric in medical imaging, as it emphasizes spatial overlap and is less sensitive to class imbalance than pixel-wise accuracy. The Hausdorff distance, which measures the maximum boundary deviation between predicted and ground truth segmentations, is often reported alongside Dice to quantify boundary quality (L. Maier-Hein et al., 2024).

The annotation cost for biomedical images is substantially higher than for natural images. Expert radiologists or pathologists must manually delineate structures slice-by-slice through 3D volumes, a process that can take hours per scan depending on the complexity and number of structures. For a typical abdominal CT scan with 200-400 slices, annotating multiple organs can require several hours of expert time, translating to annotation costs of hundreds of dollars per volume.

This extreme annotation burden makes biomedical imaging a particularly compelling application domain for AL.

2.3 Uncertainty

When selecting informative samples for understanding a system, uncertainty plays a central role. Intuitively, we expect to learn most from samples where we are uncertain about the correct response. In scientific practice, researchers commonly maintain multiple hypotheses about a system and probe samples where these hypotheses diverge, using disagreement as a proxy for uncertainty. This approach enables falsification of hypotheses and strengthens confidence in the remaining ones. Interestingly, a hypothesis may also express uncertainty when a value falls outside its defined range. However, such scenarios provide less information about which hypothesis is correct, as predictions are expected to fail in these regions.

In machine learning, uncertainty estimation is crucial beyond AL for assessing the quality of a prediction, particularly in safety-critical applications where knowing whether to trust a prediction is paramount. Consider a data generation process where response variable Y is generated from predictor variable X through model f^* and some hetereoscedatic noise $\epsilon(x)$ depent on x with a finite variance $\sigma(x)$:

$$y = f^*(x) + \epsilon(x). \tag{2.9}$$

In practice, machine learning models such as neural networks used to model f return a distribution over Y, denoted $p(y|x, \theta)$, rather than point predictions. This distributional output enables the distinction between two fundamental types of uncertainty.

This is where the notion introduced earlier about selecting informative samples in the scientific community to verify of falsify hypotheses comes into play. These notions can be described as epistemic and aleatoric uncertainty (Der Kiureghian and Ditlevsen, 2009; A. Kendall and Gal, 2017a) which describe distinct types of uncertainty that can be estimated.

Epistemic uncertainty is defined as uncertainty arising from the parameter θ due to a lack of data which can be reduced by means of more samples. Concretely this means that samples exhibiting high epistemic uncertainty allow to improve the quality of predictions.

Aleatoric uncertainty is the uncertainty arising directly from the data which is irreducible and which cannot be reduced by means of more samples. In the model described above, once the noise

term $\epsilon(x)$ has been modeled, the model is still unsure about a sample but it is not possible to predict the response more accurately.

From a logical perspective we are therefore interested in finding samples with high epistemic uncertainty in AL so that the model improves as much as possible with each queried sample.

The Bayesian thought framework allows to obtain estimates for both types uncertainty through modeling the parameter θ as random variable Θ with a probability distribution $p(\theta|\mathcal{L})$ conditioned on the observed training data $\mathcal{L}$. This framework (Mukhoti, Kirsch, et al., 2022) decomposes predictive uncertainty (PU) into epistemic uncertainty (EU) and aleatoric uncertainty (AU) as:

$$\underbrace{H[Y|x]}_{\text{PU}} = \underbrace{I[Y;\Theta|x]}_{\text{EU}} + \underbrace{\mathbb{E}_{\theta\sim\Theta}[H[Y|\theta,x]]}_{\text{AU (for i.i.d. } x)} \tag{2.10}$$

where $H[Y|x]$ is the predictive entropy (PE), $I[Y;\Theta|x]$ is the mutual information (MI) between predictions and parameters and $\mathbb{E}_{\theta\sim\Theta}[H[Y|\theta,x]$ is the expected entropy over parameters (EE) representing the AU.

The general notion of uncertainty can also be expressed by other means that do not require the Bayesian Framework and also work with point estimates of θ. From the bayesian perspective it is not possible to estimate epistemic uncertainty purely based on a point estimate of θ.

A common measure for uncertainty is the entropy $H[Y|x,\theta]$ of the models prediction $p(Y|\theta)$, this measure is highest in regions of decision boundaries as well as in regions with high aleatoric noise in the data. The maximum softmax response (Hendrycks and Gimpel, 2018) $\text{MSR} = \max_y p(y|x,\theta)$ behaves very similar in this regard.

In the case of deep learning models it is also a commonly known issue that models tend to be overconfident resulting in the emergence of calibration and other techniques (C. Guo et al., 2017).

2.4 Deep Learning

Deep learning enables end-to-end learning of hierarchical representations from data, transforming computer vision through the training of multi-layer neural networks via backpropagation and stochastic gradient descent. This section provides background on the techniques and architectures employed throughout this thesis.

The successful training of deep neural networks relies on several foundational techniques. Standard training procedures encompass loss function selection, optimizer configuration , and learning rate scheduling. Data augmentation (Y. LeCun et al., 1989; Alex Krizhevsky et al., 2017) applies transformations to training samples (rotations, flips, scaling, color jittering) to improve robustness and reduce overfitting. Weight decay (L2 regularization) (Krogh and Hertz, 1991) penalizes large parameter values to prevent overfitting. Normalization techniques, such as Batch Normalization (Ioffe and Szegedy, 2015), stabilize training by normalizing layer activations, enabling higher learning rates and faster convergence.

For uncertainty estimation based on Bayesian approaches obtaining approximate posterior distributions over model parameters is essential. The most common methods for Bayesian deep learning employ Monte Carlo Dropout (Gal and Ghahramani, 2016; Hinton et al., 2012), which treats dropout at test time as approximate Bayesian inference, or deep ensembles (Lakshminarayanan et al., 2017), which train multiple models with different initializations to capture epistemic uncertainty through predictions across the ensemble.

The remainder of this section focuses on the architectures relevant to this thesis.

2.4.1 Residual Neural Networks

The Residual Neural Network (ResNet) architecture, proposed by K. He, X. Zhang, et al. (2016), enabled the training of networks with hundreds of layers through the introduction of residual connections. The key innovation is the residual block, which learns a residual function $\mathcal{F}(x,\theta)$ that is added to the input x:

$$y = \mathcal{F}(x,\theta) + x \tag{2.11}$$

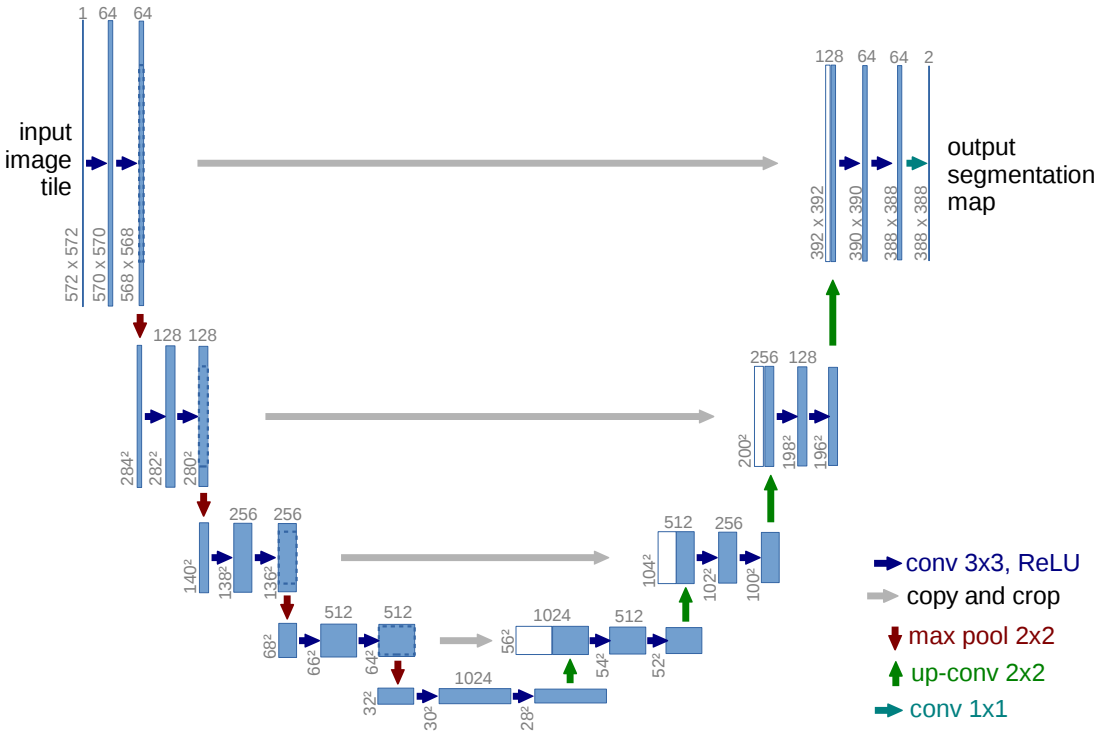


Figure 2.3: The original U-Net architecture using convolutions showing the U-shape of the encoder and decoder. Taken from Ronneberger et al. (2015)

where θ denotes the learnable parameters and y is the output. This seemingly simple modification has profound implications for gradient flow during backpropagation.

The residual connection provides a direct path for gradients to flow backward through the network via the identity mapping (x), bypassing the residual function $\mathcal{F}(x, \theta)$. During backpropagation, gradients can propagate through these skip connections without being repeatedly multiplied by weight matrices and activation functions, which typically causes gradient magnitudes to exponentially decay (vanishing gradients) or explode in very deep networks. This architectural innovation allows the training of much deeper networks than was previously possible, with ResNet models successfully trained with 50, 101, and even 152 layers—depths that would be intractable without residual connections.

The original work introduced several variants with different depths to investigate the relationship between network capacity and performance: ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152. The deeper variants (ResNet-50 and beyond) additionally employ bottleneck blocks that use 1×1 convolutions to reduce and then restore dimensionality, decreasing computational cost while maintaining representational capacity. The experiments demonstrated that deeper networks consistently achieve better performance when residual connections are present, whereas networks without such connections suffer degradation as depth increases beyond a certain point.

Since its introduction, the residual architecture has become a foundational design principle across deep learning and is still commonly used in applications (Caron et al., 2021; Radford, J. W. Kim, et al., 2021). Especially residual connections are now ubiquitously employed in state-of-the-art architectures, including Vision Transformers (Dosovitskiy et al., 2021), diffusion models (Rombach et al., 2022), and modern variants of U-Net for medical image segmentation (Isensee, T. Wald, et al., 2024). The principle has been extended beyond computer vision to natural language processing, where models like GPT (Radford, J. Wu, et al., 2019) incorporate residual connections in transformer blocks.

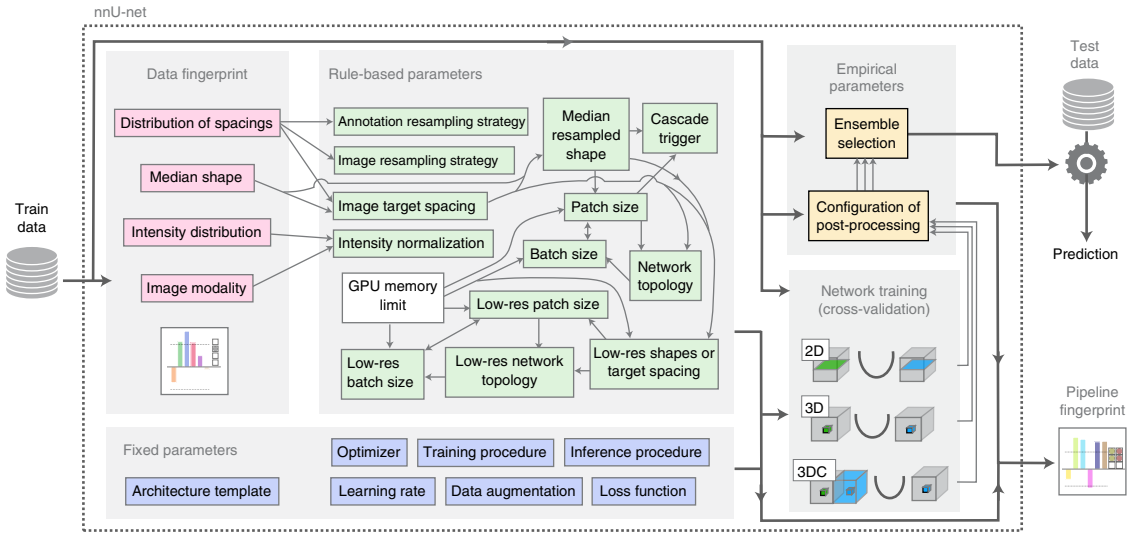


Figure 2.4: An overview of nnU-Nets adaption process to a novel dataset. Taken from Isensee, Paul F. Jaeger, et al. (2021a)

2.4.2 U-Net

The U-Net architecture was proposed by Ronneberger et al. (2015) for segmentation in biomedical images with CNNs and represents an encoder-decoder architecture which additionally introduced skip connections from encoder to decoder layers. An example of the architecture can be seen in fig. 2.3 and most importantly the architecture is agnostic to the operation meaning that the U-Net architecture is extendible to transformer models. The advantage of this approach is that information already present at a specific level of the decoder does not need to be propagated through the entire network allowing the consecutive encoder layers to learn to focus on more abstract concepts. Since its inception, the U-Net architecture is used due to these properties in many different areas including diffusion models (Esser et al., 2021) and differential equation solvers (Kidger, 2022).

In the biomedical imaging domain domain it represents to this day the state-of-the-art method for segmentation (Isensee, T. Wald, et al., 2024; Isensee, Paul F. Jaeger, et al., 2021a). Especially, the CNN based U-Nets are still generally outperforming transformer based approaches which is contrary to the trend for natural images where transformers are now more widely used than CNNs (T. Wald et al., 2025).

2.4.3 nnU-Net

While the original U-Net laid the architectural foundation for biomedical image segmentation, its practical performance in real-world applications depends heavily on a wide range of implementation and training details. The nnU-Net framework, introduced by Isensee, Paul F. Jaeger, et al. (2021a), systematically addressed this issue by automating the adaptation of U-Net configurations to new datasets. Rather than proposing a new network architecture, nnU-Net provides a set of empirically validated design rules and heuristics for preprocessing, network topology, and training, all derived from extensive experimentation across diverse biomedical segmentation tasks. Its key innovation lies in being self-configuring meaning that given a dataset, nnU-Net automatically determines appropriate normalization schemes, patch sizes, network depths, and other training hyperparameters.

For each dataset it performs a fingerprinting whose result is then used to derive the rule based parameters which are the merged with empirical fixed parameters to obtain the training recipe for the final models. These can be selected from a list of configurations which includes 3D models trained on the full resolution and lower resolution for a cascade as well as 2D models. After training, the models are then evaluated against each other to obtain the final ensemble of models. An example of this process alongside a more detailed description of the dataset fingerprint, the rule-based parameters and the fixed parameters is shwon in fig. 2.4.

This results in nnU-Net not only being a strong baseline but also a robust benchmark framework for fair comparison between methods. As a result, nnU-Net has become the de facto standard for biomedical image segmentation challenges, consistently achieving top performance across modalities and organs (Isensee, T. Wald, et al., 2024). Furthermore, its modular design and reproducible configuration logic have made it an indispensable component in the community, serving both as a practical tool for segmentation and as a methodological reference for designing and evaluating new architectures (Roy et al., 2023; J. Ma, F. Li, et al., 2024; T. Wald et al., 2025).

2.5 Active Learning

AL represents a specific case of the broader concept of Experiment Design, where the goal is to iteratively query the most relevant information from an oracle to learn specific concepts or properties of a system. The learning algorithm can actively raise questions in the form of queries, with answers provided by an oracle, enabling the algorithm to choose what it wants to learn from (Settles, 2009). The fundamental assumption of AL is that if the algorithm is allowed to be curious, it will achieve greater efficiency in the learning process with respect to required data.

Therefore fundamental assumption of AL is based on the assuption that *not all data points are equally in information to solve a given task*. This is a fundamental information theoretic assumption that is shared by many approaches in the field of machine learning, statistics, and decision-making.

The concept of queries was first introduced by Angluin (1988), who proposed a framework for learning concepts from queries in the context of formal domains, such as languages, initially exemplified in the context of playing poker learning the value of a hand of cards based on a set of possible queries.

The concept of queries was then extended to into three main types of AL (Settles, 2009):

- **Pool-based Active Learning:** The algorithm has access to a large pool of unlabeled data and selects the most informative samples to label.
- **Stream-based Active Learning:** The algorithm receives data points one at a time and decides whether to label each point based on its informativeness.
- **Membership Query Synthesis:** The algorithm can generate synthetic data points and query an oracle for their labels.

2.5.1 Active Learning Task Formulation

In the context of supervised learning, AL describes a learning paradigm where the learning algorithm actively selects which samples to label, as depicted in fig. 2.5 for pool based AL. In pool based AL, which is the focus of this thesis, the unlabeled data stems from a unchanging pool of data. Formally, we are given a dataset $\mathcal{D}$ that is divided into a labeled set $\mathcal{L}$ and an unlabeled pool $\mathcal{U}$. Initially, only a fraction of the data is labelled ("starting budget"). After initial training of the current model, the QM is used to generate queries $\mathcal{Q}_\mathcal{U}$ of a certain amount ("query size") that represent the most informative samples from $\mathcal{U}$ based on the current model predictions. Subsequently, queried samples are labeled ($\mathcal{Q}_\mathcal{L}$), moved from $\mathcal{U}$ to $\mathcal{L}$, and the model is re-trained on $\mathcal{L}$. This process is repeated until model performance is satisfying or the annotation budget has been used up completely. The goal is to select queries such that the final model achieves target performance with minimal annotation cost, or equivalently, maximizes performance for a fixed budget.

Key Components of an AL System

A practical AL system requires several components:

Query Design. The query design defines what question each query poses to the oracle. For classification, the most common query is "Which class does this sample belong to?", though alternatives exist such as "Does this sample belong to class X?" (Settles, 2009). For semantic segmentation, the query design space is considerably richer (Mittal, Tatarchenko, et al., 2019a),

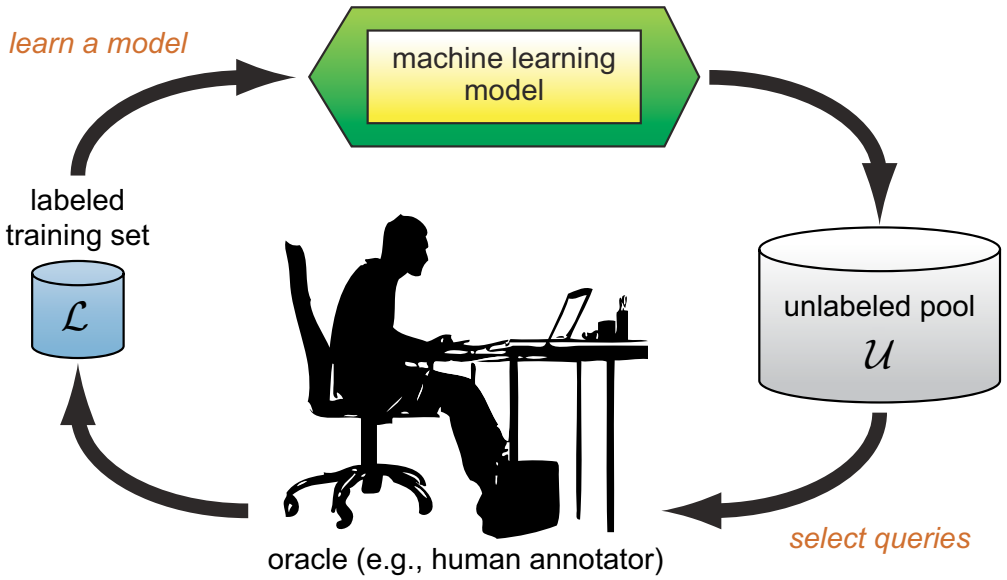


Figure 2.5: The AL loop for pool-based sampling. Starting with a small labeled set $\mathcal{L}$, the learner trains a model, applies a query strategy to select the most informative samples from the unlabeled pool $\mathcal{U}$, obtains labels from an oracle, and iterates. Image adapted from Settles (2009).

encompassing questions such as: "How should this entire image be annotated?", "What is the label for this superpixel region?", or "Which classes are present in this region?".

Query Method. A function that is used to select queries from the unlabeled pool $\mathcal{U}$. For example, a scoring function in combination, where higher scores indicate more informative queries, in combination with a maximum selection. These query methods (QMs) are commonly based on uncertainties (Gal, Islam, et al., 2017b), diversity measures (Sener and Savarese, 2018a) or a mixture of both (J. T. Ash et al., 2020; Kirsch, van Amersfoort, et al., 2019a) (detailed in section 3.1.1).

Model Configuration. The exact model used for fitting the training data $\mathcal{L}$ as different types of models behave differently on the exact same data leading to different performance. Such as changes from a convolutional neural network to a transformer architecture but already among different types of convolutional neural networks the performance can vary (Munjal et al., 2022a).

Training Paradigm. The exact training method used s.a. standard supervised, self-supervised pre-trained or semi-supervised learning (Mittal, Tatarchenko, et al., 2019a).

Starting Budget. The choice of initial labeled set $\mathcal{L}$ can significantly impact AL effectiveness (Mittal, Tatarchenko, et al., 2019a). Common approaches include random sampling, stratified sampling, or using representative examples.

Query Size. The number of queries the QM selects in each AL loop. This plays a major role in an AL system as it influences how many times the models need to be retrained to query a certain annotation budget. Further, it has been shown to strongly influence the performance of Query Methods, especially uncertainty based AL methods (Kirsch, van Amersfoort, et al., 2019a; Kirsch, Farquhar, et al., 2023).

Retraining Strategy. Whether to retrain from scratch at each iteration or fine-tune from the previous model. Full retraining is more principled but computationally expensive; incremental updates are efficient but may accumulate bias (J. Ash and Adams, 2020).

Stopping Criterion. A mechanism to determine when to stop querying. This may be based on: (i) budget exhaustion, (ii) performance saturation on a validation set, (iii) uncertainty falling below a threshold. In this thesis we will define budgets for each dataset which is most commonly used in practice Romberg et al. (2025).

Summary. These components are highly interdependent: the choice of query method may dictate optimal batch sizes, the training paradigm influences which initialization strategies are effective,

and the model architecture affects the reliability of uncertainty estimates. Throughout this thesis, we carefully consider these interactions when designing and evaluating AL systems for semantic segmentation.

The computational complexity of AL depends on several factors: the query method's complexity, the retraining cost, and the number of AL iterations. Balancing query quality with computational efficiency is a key practical consideration (Settles, 2009).

2.5.2 Active Learning in the Landscape of Adaptive Methods

AL belongs to a broader family of methods for adaptive data collection and experimental design. Understanding AL's relationship to these paradigms clarifies its unique characteristics and intellectual lineage.

Historical Context: From Statistical Experiments to Active Learning

The idea of strategically designing experiments to maximize information gain has deep roots in statistics, dating back to early 20th-century work on experimental design.

Statistical Experiment Design (Peirce, 1870s): Charles S. Peirce proposed an economic theory of scientific experimentation aimed at maximizing precision of estimates at minimal cost (Charles Sanders Peirce, 1877; Charles S Peirce, 1883). This work emphasized randomization-based inference and optimal allocation of experimental effort. Peirce articulated the core principle (Charles Sanders Peirce, 1882):

> *Logic will not undertake to inform you what kind of experiments you ought to make [...] but it will tell you how to proceed to form a plan of experimentation.*

Optimal Experiment Design (Kiefer, 1950s-1970s): This field formalizes "how to best acquire data" for statistical models through mathematically optimizable criteria. For models with parameter vector θ, optimal design minimizes the variance of parameter estimates by maximizing Fisher information $\mathcal{I}(\theta)$ for design matrix (Kiefer, 1974). Various optimality criteria (A-, D-, E-optimality) compress the information matrix differently, but all share the goal of maximizing information per observation.

Sequential Analysis (Wald, 1940s): Abraham Wald pioneered the theory of sequential hypothesis testing, where experiments are conducted one at a time and a stopping rule determines when sufficient evidence has been gathered (A. Wald, 1947). Unlike fixed-sample designs, sequential methods can achieve the same statistical power with fewer expected observations. The Sequential Probability Ratio Test (SPRT) is optimal in this sense, minimizing expected sample size for given error rates.

Sequential Design (Chernoff, Robbins, 1950s): Chernoff generalized sequential analysis to the design of experiments, where both the sample size and the choice of experiment adapt based on previous results (Chernoff, 1959). Robbins introduced the multi-armed bandit problem, formalizing the exploration-exploitation tradeoff (Robbins, 1952). These works laid the foundation for modern adaptive experimental design.

Emergence of Active Learning (1980s-1990s): The concept of queries was formalized by Angluin in the context of learning formal languages (Angluin, 1988). AL as we know it today emerged in the machine learning community, synthesizing ideas from sequential design, optimal experiment design, and computational learning theory.

Comparison with Related Paradigms

While AL shares the core assumption that not all observations are equally informative, it differs from related paradigms in its goals, constraints, and methods.

Optimal Experiment Design. In optimal design, the experimental plan (which observations to collect) is specified a priori before any data is observed. The goal is to choose a design matrix X that maximizes Fisher information for a known parametric model. In contrast, AL selects observations iteratively based on the evolving model. Furthermore, optimal design commonly assumes the model form is known; AL assumes only a hypothesis class and learns the mapping from data.

Sequential Analysis. Sequential analysis adaptively decides *when to stop* collecting data, but the type of data collected at each step is typically fixed (e.g., independent draws from the same distribution). AL, on the other hand, adaptively decides *what data to collect* but typically operates under a fixed budget. Both address sample efficiency, but from complementary angles.

Multi-Armed Bandits. Bandits and AL both face exploration-exploitation tradeoffs. However, bandits optimize *cumulative reward* over a sequence of actions, whereas in AL we are in practice interested in the *final model's performance* after a fixed number of queries. In a bandit problem, early mistakes are commonly penalized; whereas when AL is used in practice, only the final model matters. Nonetheless, some bandit problems can be formalized with AL (Srinivas et al., 2010).

Reinforcement Learning. Reinforcement learning learns a policy that maximizes cumulative reward over sequential decisions, often with delayed feedback and complex state-action dynamics. While some Reinforcement learning problems involve learning from strategically collected data (exploration strategies), RL's goal is policy optimization, not supervised model training. However, Reinforcement learning techniques (e.g., value functions, policy gradient) have inspired AL query strategies (Fang et al., 2017).

Key Insights from Related Paradigms

Despite their differences, these paradigms offer valuable insights for AL:

1. **Information measures are central**: Optimal experiment design's use of Fisher information and entropy-based criteria has directly influenced AL query strategies (Gal, Islam, et al., 2017b; Kirsch, 2024).
2. **Sequential adaptation is powerful**: Sequential analysis demonstrated that adaptive methods can achieve the same goals with fewer samples, validating AL's core premise.
3. **Exploration-exploitation tradeoffs are fundamental**: Bandit algorithms' principled handling of exploration vs. exploitation can inform how AL should balance uncertainty sampling (exploitation) with diversity (exploration).

Understanding AL within this broader landscape highlights both its unique contributions and the intellectual debts it owes to decades of research in adaptive experimental design.

2.5.3 Active Learning Evaluation

The evaluation of AL methods is typically conducted using a key performance metric—such as accuracy, Dice score, or mean IoU—on a held-out test set. During each iteration of the AL loop, this performance metric is recorded, resulting in a time-series-like performance trajectory over the annotation budget. Various evaluation methods have been proposed to summarize and compare these trajectories. The following provides an overview of the most commonly used approaches.

Visual Analysis. A widely adopted and intuitive approach (K. Kim et al., 2021; Mittal, Tatarchenko, et al., 2019a; Gal, Islam, et al., 2017b; Kirsch, van Amersfoort, et al., 2019b; Kirsch, Farquhar, et al., 2023) consists of plotting the performance metric against the annotation budget and analyzing the resulting curves visually. This allows for straightforward comparison between a small number of QMs and directly reveals at which annotation budget a particular method performs best. However, visual analysis can be time-consuming and subjective, especially when comparing many QMs. Furthermore, without confidence intervals or statistical markers, visual interpretation may obscure significant differences between methods.

Area Under the Budget Curve (AUBC). The Area Under the Budget Curve (AUBC) (Zhan, H. Liu, et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B. Chan, 2022b) summarizes the overall AL performance into a single scalar value by integrating the performance curve across all annotation budgets, typically using the trapezoidal rule. By aggregating the complete trajectory, the AUBC provides a compact and easily interpretable comparison between multiple QMs. Its main limitation lies in its inability to distinguish where along the budget axis improvements occur, as early and late iterations contribute equally to the total area.

Final Performance. The Final Performance metric considers only the performance achieved in the final iteration of the AL loop, thereby disregarding intermediate results. Its appeal lies in its simplicity and its direct interpretability relative to the fully supervised baseline trained on the complete dataset. However, by design this metric does not capture earlier iterations where two QMs might achieve identical final performance, even though one reached high accuracy much earlier in the annotation process. For this reason, it is often used in combination with the AUBC as the combination allows insights in where improvements occur, as the previous example could be distinguished by means of the AUBC (Zhan, H. Liu, et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B. Chan, 2022b).

Annotation Budget to Reach a Target Performance. This metric quantifies the annotation budget required for a QM to achieve a predefined target performance (commonly 95% or 99% of the full-dataset performance) (Kirsch, van Amersfoort, et al., 2019b; Gal, Islam, et al., 2017b). It provides an interpretable scalar indicating label efficiency: the lower the required budget, the more effective the AL method. However, its validity strongly depends on the choice of the target performance, which must be appropriately justified for the task at hand. Additionally, for large query sizes, estimating the exact budget at which the target performance is reached becomes increasingly imprecise due to coarse annotation steps.

Pairwise Penalty Matrix (PPM) The Pairwise Penalty Matrix (PPM) (J. T. Ash et al., 2020) evaluates QMs by comparing their trajectories pairwise using a two-sided t-test (commonly with a significance level of $\alpha = 0.05$) at each annotation budget. This results in a matrix where each row and column correspond to a QM; an entry indicates how often the method in row i significantly outperforms the one in column j. High values in a QM's row and low values in its column thus signify strong performance. A "mean row" is often included to summarize how frequently each QM is outperformed by others. The PPM offers a convenient way to aggregate results across multiple datasets or experiments while preserving statistical comparability. However, it only captures win-loss relationships and not the magnitude of differences. This means that it cannot distinguish between marginal and substantial improvements once the difference is significant.

Overview All of the aforementioned evaluation methods inherently depend on how the annotation budget is defined. In object recognition tasks such as image classification, the annotation budget is typically expressed as the number of labeled images, which directly corresponds to the number of queries. This formulation implicitly assumes that each image requires a comparable amount of annotation effort. However, in tasks like semantic segmentation, this assumption becomes problematic: queries may correspond to entire images, regions within images, or even individual pixels, each of which entails vastly different annotation costs. The manual effort required to create accurate segmentation masks strongly depends on the complexity and structure of the

depicted content. Moreover, as noted by Settles (2011), AL methods often prioritize complex or ambiguous samples, which may further increase the annotation effort per query. Therefore, equating annotation cost across queries, or assuming it remains identical to that in non-AL settings, can lead to misleading evaluations of AL efficiency.

Chapter 3

State-of-the-Art

> We can only see a short distance ahead, but we can see plenty there that needs to be done.
>
> Alan Turing

In this section, we will discuss the current state-of-the-art for annotation efficient learning with respect to Active Learning as well as other related paradigms on a concept level. Detailed discussions of the related work necessary to set our contributions into context are provided for each chapter 5, chapter 6, chapter 7 and chapter 8 separately.

3.1 Active Learning

3.1.1 Query Methods

In pool-based Active Learning, the Query Method (QM) determines based on the model $p(Y|x, \theta)$ trained on the current labeled set $\mathcal{L}$ which samples from the unlabeled pool $\mathcal{U}$ are selected for annotation. The choice of query method fundamentally determines AL's effectiveness, as different methods embody different strategies for identifying informative samples.

Query methods can be broadly categorized into two main families (Settles, 2009): **Uncertainty-based methods** exploit model predictions to query samples where the model is most uncertain, aiming to refine decision boundaries efficiently. **Diversity-based methods** aim to cover the input distribution broadly, querying representative samples that explore different regions of the feature space to avoid redundant queries. Recent work has explored hybrid approaches that balance both objectives (J. T. Ash et al., 2020).

This section introduces the QMs evaluated in this thesis in the context of classification, representing key approaches from each category. The extension of these QMs to semantic segmentation is discussed in section 3.1.2. These methods were selected based on their prominence in recent AL literature, their relevance to semantic segmentation tasks, and their representation of different query selection philosophies—from simple uncertainty sampling to sophisticated Bayesian approaches to geometric coverage methods. For comprehensive surveys of AL methods beyond those covered here, we refer readers to Settles (2009) and P. Liu et al. (2022).

Notation Whenever possible, we will define a query method by means of its query score $s(x)$ for sample x, where higher scores indicate higher priority for annotation. The query size K declares how many samples are queried in each AL iteration. For batch methods, $s(x_1, ..., x_K)$ scores entire batches jointly. We use $\mathcal{L}$ to denote the labeled set and $L(\cdot)$ to denote the loss function to avoid notational conflicts and $\mathcal{U}$ refers to the unlabeled pool.

Baseline Method

Random Random sampling draws samples from the unlabeled pool $\mathcal{U}$ uniformly at random, serving as the fundamental baseline for AL evaluation. While it does not leverage model information, random sampling provides unbiased coverage of the data distribution $p(x)$ and serves as an exploratory strategy. Strictly speaking, it is not necessarily a QM, but we will treat it as such.

Uncertainty-Based Methods

Uncertainty-based methods query samples where the model exhibits high prediction uncertainty, operating under the principle that correcting the model on uncertain samples will most efficiently improve decision boundaries.

Entropy Entropy sampling queries samples with the highest prediction entropy, measuring uncertainty in the model's output distribution. For a sample x and model with C classes, the entropy score is:

$$s(x) = H[Y|x,\theta] = -\sum_{c=1}^{C} p(Y=c|x,\theta)\log p(Y=c|x,\theta) \tag{3.1}$$

Higher entropy indicates higher uncertainty; samples with maximum entropy are selected. Entropy sampling represents a straightforward uncertainty-based approach that requires only a single forward pass per sample, making it computationally efficient. However, it does not distinguish between samples where the model is uncertain due to lack of knowledge versus inherent label ambiguity. Samples are selected greedily based on highest scores (top-K sampling).

When using a Bayesian model, the predictive entropy is computed by first averaging predictions across the posterior distribution over model parameters $H[Y|x,\mathcal{L}] = H[\mathbb{E}_{\theta\sim p(\theta|\mathcal{L})}[p(Y|x,\theta)]]$.

BALD Bayesian Active Learning by Disagreement (BALD) selects samples that maximize the mutual information between the predicted labels Y and model parameters Θ, effectively measuring epistemic (model) uncertainty (Gal, Islam, et al., 2017a). The BALD score is:

$$s(x) = I[Y;\Theta|x,\mathcal{L}] = H[Y|x,\mathcal{L}] - \mathbb{E}_{p(\theta|\mathcal{L})}[H[Y|x,\theta]] \tag{3.2}$$

where $p(Y|x,\mathcal{L}) = \mathbb{E}_{p(\theta|\mathcal{L})}[p(Y|x,\theta)]$ is the predictive distribution. This decomposes into the predictive entropy minus the expected conditional entropy, isolating epistemic uncertainty. In practice, the expectation over $p(\theta|\mathcal{L})$ is approximated using Monte Carlo dropout (Gal, Islam, et al., 2017a), sampling M different dropout masks during inference. BALD typically requires $M = 50-100$ forward passes per sample, making it more computationally expensive than single-pass methods like Entropy, but it has demonstrated strong performance particularly when epistemic uncertainty is high. Samples are selected greedily based on highest scores (top-K sampling).

PowerBALD PowerBALD addresses a limitation of BALD: top-K sampling of BALD scores often selects highly similar samples, leading to correlated queries that provide redundant information (Kirsch, Farquhar, et al., 2023). PowerBALD perturbs BALD scores on a logarithmic scale with Gumbel noise:

$$s(x) = \log(I[Y;\Theta|x,\mathcal{L}]) + \varepsilon \tag{3.3}$$

where $\varepsilon \sim \text{Gumbel}(0,\beta^{-1})$. The parameter β controls the exploration-exploitation trade-off: larger values of β preserve the original BALD ranking (exploitation), while smaller values introduce more randomness (exploration). As $\beta \to \infty$, PowerBALD reduces to standard BALD; as $\beta \to 0$, selection becomes random. This stochastic sampling encourages diversity in batch selection while still favoring high-scoring samples. Kirsch, Farquhar, et al. (2023) note that query correlation naturally decreases in later AL rounds as the model becomes more confident, making the diversity mechanism most valuable in early rounds.

PowerPE PowerPE applies the PowerBALD approach to predictive entropy (Kirsch, Farquhar, et al., 2023), perturbing entropy scores with Gumbel noise:

$$s(x) = \log(H[Y|x, \theta]) + \varepsilon \tag{3.4}$$

where $\varepsilon \sim \text{Gumbel}(0, \beta^{-1})$. This provides an alternative to PowerBALD which does not necessarily requires a a Bayesian model. While still being able to be applied in the Bayesian setting as described earlier for the Entropy Query Method by using $H[Y|x, \mathcal{L}]$.

SoftrankBALD SoftrankBALD provides an alternative approach to addressing BALD's correlation problem by perturbing the rankings rather than the scores directly (Kirsch, Farquhar, et al., 2023):

$$s(x) = -\log(r(x)) + \varepsilon \tag{3.5}$$

where $r(x)$ is the rank of sample x when sorted by BALD score (lowest rank = highest BALD score) and $\varepsilon \sim \text{Gumbel}(0, \beta^{-1})$ with the same interpretation of β as in PowerBALD. By operating on ranks, SoftrankBALD is less sensitive to the absolute scale of BALD scores, which can vary across AL rounds. The negative logarithm transforms ranks into a score where higher values indicate better queries, and the Gumbel noise introduces controlled stochasticity.

Diversity-Based Methods

Diversity-based methods prioritize geometric coverage of the feature space rather than model uncertainty, selecting samples that are representative of unexplored regions.

Core-Set Core-Set frames AL as a geometric coverage problem: find a subset of samples that best approximates the full dataset in feature space (Sener and Savarese, 2018b). Specifically, Core-Set queries samples in form of a Set $\mathcal{S} = x_1, ..., x_K$ that minimize the maximum distance from any unlabeled sample to its nearest labeled sample in the model's representation space:

$$\min_{\mathcal{S} \subset \mathcal{U}} \max_{x \in \mathcal{U} \setminus \mathcal{S}} \min_{x' \in \mathcal{L} \cup \mathcal{S}} \|f(x) - f(x')\|_2 \tag{3.6}$$

where $f(x)$ denotes the representation (typically from the penultimate layer) of sample x. This objective corresponds to the K-center problem, which is NP-hard. Core-Set is commonly implemented using the K-center greedy algorithm (Yoo and Kweon, 2019a), which iteratively selects the unlabeled sample farthest from the current labeled set, achieving a 2-approximation guarantee. This approach tends to select samples from the tails and diverse regions of the distribution, providing good coverage. Core-Set is classified as an exploratory strategy and requires computing pairwise distances in feature space.

Hybrid Methods

Hybrid methods combine uncertainty and diversity objectives, attempting to balance informativeness with geometric coverage.

BADGE Batch Active learning by Diverse Gradient Embeddings (BADGE) combines uncertainty and diversity by clustering gradient embeddings (J. T. Ash et al., 2020). For each sample $x \in \mathcal{U}$, BADGE computes the gradient of the loss with respect to the final layer parameters:

$$\mathcal{G} = \{\nabla_{\theta_{-1}} L(\hat{y}(x), p(Y|x, \theta)) \mid x \in \mathcal{U}\} \tag{3.7}$$

where θ_{-1} denotes the parameters of the final layer, $L(\cdot)$ is the loss function, and $\hat{y}(x)$ is the predicted label (typically the argmax prediction). These gradient embeddings capture both uncertainty (samples with high loss gradients) and diversity (samples with different gradient directions indicate diverse parameter updates). BADGE then applies K-means++ initialization to select K centers from $\mathcal{G}$, ensuring both high gradients and diverse directions.

Kirsch and Gal (2022)[1] provide theoretical justification for BADGE, showing that this selection criterion is equivalent to maximizing the joint mutual information of selected samples and their hard pseudo-labels under an uninformative posterior, based on connections between Fisher information and mutual information.

3.1.2 Query Methods for Semantic Segmentation

For semantic segmentation segmentation of 3D biomedical images the effectiveness of the QMs introduced in section 3.1.1 has been less thoroughly studied compared to image classification. Pixel-level predictions introduce unique challenges: uncertainty may be spatially correlated, diversity should account for both image-level and pixel-level variation, and computational costs scale with image resolution.

The label y now carries the same spatial resolution as its corresponding image x. This leads to the query design allowing either pixels, super-pixels, patches or entire images (Mackowiak et al., 2018; Mittal, Tatarchenko, et al., 2019a).

QMs that operate based on uncertainty-Entropy or BALD-can be easily applied to all of these query designs by means of an aggregation strategy, such as the mean uncertainty over the corresponding region (Mackowiak et al., 2018; Mittal, Tatarchenko, et al., 2019a; B. Xie et al., 2022).

QMs that require representations such as Core-Set have also been applied to semantic segmentation (Mittal, Tatarchenko, et al., 2019a; Burmeister et al., 2022; Föllmer et al., 2024). This has, however, only been done with query designs on image level where one forward pass of the model exactly covers one forward pass. The reason is that representation based methods are highly dependent on the quality of representations and it is not necessarily generally clear what exactly the representation for a part of an image is. This question is therefore especially detrimental for querying parts of an image in semantic segmentation.

Hybrid QMs such as BADGE also have been mostly adapted to similar scenarios (Aklilu and Yeung, 2022; Föllmer et al., 2024) which is because the computational complexity of these methods scales much more poorly for larger pools of potential queries.

Concluding, compared to other QMs uncertainty-based QMs are highly adaptable to different query designs for semantic segmentation. This is because adapting them only necessitates an appropriate aggregation strategy. The effect of an aggregation strategy will be further analyzed in chapter 6.

3.2 Alternative paradigms for annotation efficient learning

In this section we will introduce the following supervised learning paradigms that are also commonly employed in deep supervised learning to reduce annotation effort.

3.2.1 Transfer Learning

The main idea of transfer learning is to leverage knowledge learned from a source task or domain to improve performance on a related target task or domain, particularly when labeled data for the target task is limited (Pan and Q. Yang, 2009; Weiss et al., 2016). Rather than training models from random initialization, transfer learning exploits pre-trained models that have already learned useful representations from large-scale datasets, enabling faster convergence and better generalization with fewer labeled samples. The underlying assumption is that features learned on the source task capture general patterns that are transferable to the target task, such as low-level visual features (edges, textures) or mid-level semantic concepts (object parts, shapes). As the model begins training with informative initialization rather than random weights, it requires fewer labeled annotations for the target task to achieve comparable or superior performance to training from scratch. In the domain of computer vision, common transfer learning strategies include:

- **Feature extraction:** Using a pre-trained model as a fixed feature extractor, where only

[1] See proposition 7.2 and 6.2

the final classification layer is trained on the target task while earlier layers remain frozen (Sharif Razavian et al., 2014; Donahue et al., 2014)

- **Fine-tuning:** Initializing a model with pre-trained weights and continuing training on the target task, allowing all or some layers to adapt to the new domain (Yosinski et al., 2014; M. Long et al., 2015)
- **Domain adaptation:** Explicitly reducing the distribution shift between source and target domains through techniques such as domain adversarial training or distribution alignment (Ganin et al., 2016; Tzeng et al., 2017)
- **Multi-task learning:** Training a single model on multiple related tasks simultaneously, enabling knowledge sharing across tasks through shared representations (Caruana, 1997; Ruder, 2017)
- **Few-shot learning:** Learning to learn from few examples by training on many related tasks, enabling rapid adaptation to new tasks with minimal labeled data (Finn et al., 2017; Snell et al., 2017)
- **Pre-training on large-scale datasets:** Training models on massive datasets such as ImageNet, then transferring to downstream tasks with limited data (Deng et al., 2009; Ridnik et al., 2021)
- **Foundation models:** Leveraging very large models pre-trained on diverse data at scale, which exhibit strong zero-shot and few-shot transfer capabilities across numerous tasks (Bommasani et al., 2021; Kirillov et al., 2023; Radford, J. W. Kim, et al., 2021)

In recent years, foundation models pre-trained on large-scale diverse datasets have emerged as the dominant paradigm for transfer learning, with models such as CLIP (Radford, J. W. Kim, et al., 2021), Segment Anything Model (SAM) (Kirillov et al., 2023), and DINOv2 (Oquab et al., 2024) representing state-of-the-art approaches that demonstrate remarkable transfer capabilities across diverse computer vision tasks. The effectiveness of transfer learning comes with several practical requirements and trade-offs. Transfer learning requires that the source and target domains share sufficient similarity in their underlying structure and feature distributions, as large domain gaps can limit the transferability of learned representations. Additionally, accessing and storing large pre-trained models requires significant computational resources and storage capacity, with modern foundation models often containing billions of parameters. Furthermore, fine-tuning pre-trained models on small target datasets carries the risk of overfitting, requiring careful regularization and hyperparameter tuning to balance retention of pre-trained knowledge with adaptation to the new task. However, these costs are substantially outweighed by transfer learning's ability to dramatically reduce annotation requirements and training time. Pre-trained models provide strong initialization that often achieves competitive performance with orders of magnitude fewer labeled samples than training from scratch (K. He, Girshick, et al., 2019). This makes transfer learning a foundational component of modern computer vision pipelines, where it serves as the default starting point for many practical applications (Neyshabur et al., 2020).

3.2.2 Self-Supervised Learning

The main idea of self-supervised learning is to pre-train a model on so called pretext tasks, that are independent of annotations, to learn representations that generalize well to one or more specific downstreamtasks (Gui et al., 2024). After the model has been pre-trained in a self-supervised fashion, it is then commonly finetuned on specific downstram tasks in a supervised fashion. As the pre-trained model has already learned well generalizing representations, the model then requires fewer annotated samples during the supervised finetuning then if it was trained from a random initialization. The most influential pre-training paradigms are:

- **Colorization:** Predicting color channels from grayscale images, forcing the model to understand object semantics (Richard Zhang et al., 2016)
- **Context prediction:** Reconstructing missing image regions, learning spatial relationships and object structure (Pathak et al., 2016; K. He, X. Chen, et al., 2022)

- **Jigsaw puzzles:** Predicting correct arrangements of shuffled image patches, learning part-whole relationships (Noroozi and Favaro, 2017)
- **Rotation prediction:** Identifying image rotation angles, learning orientation-invariant features (Gidaris et al., 2018)
- **Contrastive learning:** Maximizing agreement between augmented views of the same image while distinguishing different images (K. He, H. Fan, et al., 2020; T. Chen et al., 2020)
- **Knowledge distillation:** Distilling knowledge from teacher networks into student networks through self-distillation mechanisms (Grill et al., 2020; Caron et al., 2021; Oquab et al., 2024; Siméoni et al., 2025)
- **Generative modeling:** Learning representations through image generation using Generative Adversarial Networks or Autoencoders (X. Liu et al., 2021)

In recent years, contrastive learning has emerged as the dominant SSL paradigm for classification tasks, with SimCLR (T. Chen et al., 2020) and most notably DINO (Caron et al., 2021) representing state-of-the-art approaches that are widely employed across computer vision applications. The effectiveness of SSL comes with several practical requirements and trade-offs. SSL pre-training demands access to large unlabeled datasets—often orders of magnitude larger than typical supervised training sets—as representation quality improves with dataset scale. Additionally, the computational overhead of SSL exceeds that of supervised training: pre-training requires processing extensive unlabeled data through numerous epochs while solving computationally intensive pretext tasks. However, this upfront investment only needs to be made once as upon the emergence of new annotated data, only the finetuning needs to be performed, and can be further justified by SSL's strong transfer learning properties. Pre-trained SSL models consistently demonstrate robust generalization to multiple datasets not encountered during training, enabling a single pre-training phase to benefit numerous downstream tasks (Gui et al., 2024). This makes SSL particularly economical in settings where models will be deployed across multiple related problems, effectively amortizing the pre-training cost across applications.

3.2.3 Semi-Supervised Learning

The main idea of semi-supervised learning is to leverage both labeled and unlabeled data during training to improve model performance beyond what can be achieved with labeled data alone (Van Engelen and Hoos, 2020). Semi-supervised learning exploits unlabeled data directly during the training of the target task. By incorporating assumptions about the relationship between the data distribution and the decision boundary, such as smoothness, cluster, or manifold assumptions, semi-supervised methods can effectively utilize the abundant unlabeled data to regularize the model and improve generalization (Chapelle et al., 2006). As the model learns from both labeled samples in $\mathcal{L}$ and the structure of unlabeled samples in $\mathcal{U}$ simultaneously, it requires fewer labeled annotations than purely supervised learning to achieve comparable performance. In the domain of computer vision, prominent semi-supervised learning approaches include:

- **Consistency regularization:** Enforcing consistent predictions for different augmentations of the same unlabeled image, encouraging the model to learn robust and invariant representations (Bachman et al., 2014; Laine and Aila, 2017; Miyato et al., 2018)
- **Pseudo-labeling:** Assigning high-confidence predictions on unlabeled data as pseudo-labels and using them as additional training targets, iteratively improving the model (D.-H. Lee et al., 2013; Q. Xie et al., 2020)
- **Hybrid approaches:** Combining consistency regularization with pseudo-labeling to leverage both mechanisms, achieving state-of-the-art performance (Sohn, Berthelot, C.-L. Li, et al., 2020; B. Zhang, Yidong Wang, et al., 2021; Yidong Wang et al., 2022)
- **Co-training and multi-view learning:** Training multiple models or views that teach each other on unlabeled data, exploiting complementary information from different perspectives (Blum and Mitchell, 1998; Qiao et al., 2018)

- **Entropy minimization:** Encouraging the model to make confident predictions on unlabeled data by minimizing prediction entropy, pushing decision boundaries away from high-density regions (Grandvalet and Bengio, 2004)
- **Graph-based methods:** Propagating labels through similarity graphs constructed from the data, assuming that nearby points in feature space should have similar labels (X. Zhu et al., 2003; Iscen et al., 2019)
- **Generative models:** Using generative models such as Variational Autoencoders or Generative Adversarial Networks to model the joint distribution of data and labels (Kingma et al., 2014; Salimans et al., 2016)

In recent years, hybrid approaches combining consistency regularization and pseudo-labeling have emerged as the dominant paradigm for semi-supervised learning (Oliver et al., 2019), with FixMatch (Sohn, Berthelot, C.-L. Li, et al., 2020), FlexMatch (B. Zhang, Yidong Wang, et al., 2021), and FreeMatch (Yidong Wang et al., 2022) representing state-of-the-art methods that are widely employed across computer vision applications. The effectiveness of semi-supervised learning comes with several practical requirements and trade-offs. Semi-supervised learning requires that the unlabeled data distribution is relevant to the task and similar to the labeled data distribution, as distribution mismatch can degrade performance rather than improve it. Additionally, semi-supervised methods increase computational cost compared to purely supervised training, as they require processing the unlabeled pool $\mathcal{U}$ in each training iteration and computing auxiliary losses such as consistency or entropy terms. Furthermore, semi-supervised learning introduces additional hyperparameters (s.a. weight of the unsupervised loss, confidence thresholds for pseudo-labeling, and augmentation strategies) that must be tuned, typically requiring a validation set.

3.2.4 Weakly Supervised Learning

The main idea of weakly supervised learning is to train models using weak or imperfect supervision that is cheaper or easier to obtain than full annotations, while achieving performance comparable to fully supervised methods (Z.-H. Zhou, 2018; T. Zhang et al., 2021). As weakly supervised learning is especially useful in domains with a high annotation cost per image, we will focus on semantic segmentation in this section.

Rather than requiring expensive pixel-level annotations for every training sample, weakly supervised learning exploits alternative forms of supervision such as image-level labels, bounding boxes, scribbles, or points, which can be obtained at a fraction of the cost and time. The underlying principle is that models can learn to infer the full ground truth structure from incomplete or noisy supervision by leveraging assumptions about the data or incorporating additional constraints and regularization. As weak annotations require substantially less annotation effort per sample than full supervision, models can be trained on larger datasets or achieve comparable performance with reduced annotation budgets.

In the domain of semantic segmentation, where pixel-level annotations are particularly expensive and time-consuming to obtain, weakly supervised learning has emerged as a crucial research direction. Common forms of weak supervision for semantic segmentation include:

- **Image-level labels:** Using only class labels indicating which object categories are present in the image, without specifying their locations or boundaries (Ahn and Kwak, 2018; Yude Wang et al., 2020)
- **Bounding boxes:** Providing rectangular boxes around objects of interest, which are faster to annotate than pixel-level masks but lack precise boundary information (Dai et al., 2015; Khoreva et al., 2017; Song et al., 2019)
- **Scribbles:** Annotating objects with sparse lines or strokes drawn on or near the objects, capturing rough location and shape with minimal effort (D. Lin et al., 2016; Tang et al., 2018; B. Zhang, Yao, et al., 2020)
- **Points:** Marking objects with single points or sparse point sets, representing the minimal supervision while still indicating object presence and approximate location (Bearman et al., 2016; Laradji et al., 2021)

- **Patches:** Annotating selected patches or image regions rather than entire images, providing localized supervision that reduces annotation burden while maintaining detailed information within annotated regions (G. Lin et al., 2016; Papandreou et al., 2015)
- **Grids and superpixels:** Annotating at coarser granularity using regular grids or superpixel-based regions, reducing annotation time while maintaining some spatial structure (G. Lin et al., 2016; Vernaza and Chandraker, 2017)
- **Class activation maps (CAMs):** Leveraging attention mechanisms in classification networks to identify discriminative regions, which can be used as pseudo-labels for segmentation (B. Zhou et al., 2016; Ahn and Kwak, 2018; J. Lee et al., 2021)
- **Noisy or crowdsourced labels:** Using annotations from non-expert annotators or automated systems that may contain errors, trading annotation quality for quantity or cost (Rodrigues and Pereira, 2018; Tanno et al., 2019)

In recent years, approaches combining image-level labels with class activation maps have emerged as a dominant paradigm for weakly supervised semantic segmentation, with methods such as AffinityNet (Ahn and Kwak, 2018), IRNet (Ahn, Cho, et al., 2019), and SEAM (Yude Wang et al., 2020) representing state-of-the-art techniques that generate high-quality pseudo-labels from weak supervision. More recently, foundation models such as the Segment Anything Model (SAM) (Kirillov et al., 2023) have enabled new weakly supervised approaches by providing strong segmentation priors that can be guided with minimal supervision such as points or boxes.

The effectiveness of weakly supervised learning comes with several practical requirements and trade-offs. Weakly supervised methods typically achieve lower performance than fully supervised counterparts when the same amount of annotation budget is spent, as weak annotations contain less information per sample. Additionally, weakly supervised approaches often require more complex training procedures, including multi-stage training, pseudo-label generation and refinement, or sophisticated regularization techniques to compensate for incomplete supervision. Furthermore, the choice of weak supervision type involves a fundamental trade-off between annotation cost and information content: image-level labels are cheapest but least informative, while scribbles or boxes provide more guidance at higher cost.

However, these limitations are outweighed by weakly supervised learning's ability to dramatically reduce total annotation costs when leveraging the cost-quality trade-off appropriately. For semantic segmentation, pixel-level annotation can take several minutes per image, while image-level labels require only seconds, and bounding boxes or scribbles fall in between (Bearman et al., 2016; D. Lin et al., 2016). This cost difference means that weakly supervised methods can often achieve better performance than fully supervised methods when comparing equal annotation budgets rather than equal numbers of samples, as a practitioner can potentially annotate $10 - 100\times$ more images with weak supervision for the same cost as full supervision.

This makes weakly supervised learning a particularly attractive option for semantic segmentation tasks where pixel-level annotation is prohibitively expensive and alternative annotation types can be obtained more efficiently (T. Zhang et al., 2021).

3.2.5 Comparative Analysis of Annotation Efficient Learning

We previously introduced four distinct paradigms for improving learning efficiency beyond standard supervised learning: transfer learning, self-supervised learning, semi-supervised learning, and weakly supervised learning. While each addresses the challenge of reducing annotation requirements, they operate through fundamentally different mechanisms and at different stages of the learning pipeline. This section provides a unified comparative analysis of these approaches and examines their relationship to AL.

These annotation-efficient learning paradigms can be characterized along three key dimensions that capture their fundamental mechanisms:

Dimension 1: What is leveraged? This dimension describes the primary resource each paradigm exploits:

- **Transfer learning:** Leverages knowledge from external source tasks or domains, typically through pre-trained models trained on large-scale datasets
- **Self-supervised learning:** Leverages the structure and patterns within unlabeled data from the target domain through pretext tasks
- **Semi-supervised learning:** Leverages the distribution and manifold structure of unlabeled data alongside labeled data during training
- **Weakly supervised learning:** Leverages alternative, cheaper forms of supervision (image labels, boxes, scribbles) instead of full annotations
- **Active Learning:** Leverages model uncertainty and data characteristics to strategically select which instances to annotate

Dimension 2: When does it operate? This dimension identifies the stage in the learning pipeline where each paradigm operates:

- **Transfer learning:** Operates in the initialization phase, providing pre-trained weights before target task training
- **Self-supervised learning:** Operates in a pre-training phase, learning representations before supervised fine-tuning
- **Semi-supervised learning:** Operates during training, simultaneously leveraging labeled and unlabeled data
- **Weakly supervised learning:** Operates during training, learning from weak annotations and generating pseudo-labels
- **Active Learning:** Operates in the data selection phase, determining which samples to annotate across multiple rounds

Dimension 3: What cost does it address? This dimension clarifies which aspect of annotation cost each paradigm reduces:

- **Transfer learning:** Reduces the number of labeled samples needed by providing informative initialization
- **Self-supervised learning:** Reduces the number of labeled samples needed by learning from unlabeled data first
- **Semi-supervised learning:** Reduces the number of labeled samples needed by exploiting unlabeled data during training
- **Weakly supervised learning:** Reduces the cost per annotation by accepting cheaper, less detailed supervision
- **Active Learning:** Reduces the number of labeled samples needed by selecting the most informative instances

Orthogonality. A critical observation is that these paradigms are largely *orthogonal* to one another, meaning they address different aspects of the learning problem and can be largely combined without conflict. Potentially, this allows them to be used in any combination with each other as long as the benefit of additionally incorporating them outweighs the additional cost incurred.

Verifiability during deployment Whether practitioners can validate empirically whether a paradigm leads to benefits for their specific problem is critical as it reduces the necessary trust into the validation:

- **Transfer learning:** *Yes*—compare pre-trained versus random initialization on the same labeled set.
- **Self-supervised learning:** *Yes*—compare with versus without SSL pre-training on the same labeled set.
- **Semi-supervised learning:** *Yes*—compare semi-supervised versus purely supervised training on the same labeled set.
- **Weakly supervised learning:** *Partially*—annotation time reduction is measurable, but validating the performance-cost trade-off requires obtaining both weak and full annotations for comparison.
- **Active Learning:** *No*—validating query strategies requires labeling multiple AL trajectories, contradicting AL's purpose. We discuss this in further detail in section 4.2.

Chapter 4

Theoretical Considerations for Active Learning

> The first principle is that you must not fool yourself and you are the easiest person to fool.
>
> Richard Feynman

Research Question: What are general aspects necessary for the evaluation of Active Learning methods?

When practitioners consider deploying AL in a real-world project, they face a fundamental decision: Will the investment in AL methodology reduce overall project costs sufficiently to justify its adoption? This section formalizes this decision problem and establishes requirements for evaluation frameworks that can inform such decisions. The results of this chapter are used to guide the design of the evaluation frameworks in chapter 5 and chapter 7 as well as the methodological design in chapter 8.

Unlike standard machine learning, where practitioners can validate model configurations using fixed labeled datasets, AL operates in a setting where the very data to be labeled is being strategically selected. This creates unique challenges for validation and evaluation that we explore in detail.

4.1 Economic Framework for Active Learning

In its very essence the question of whether to employ AL or not, is an economical question as its main purpose lies in reducing annotation cost (Settles, 2011). In a strongly simplified manner, the total cost of a machine learning project using Active Learning (C_{AL}) can be expressed as:

$$C_{\text{AL}} = C_{\text{baseline}} - r \cdot c_{\text{annotation}} + c_{\text{AL-overhead}} \tag{4.1}$$

where each of these components is influenced by multiple factors:

Baseline cost (C_{baseline}). The baseline cost is the cost of the project using a standard approach (e.g., random sampling with a predetermined annotation budget). It is determined by the task complexity, model selection, data characteristics, annotation methodology, and infrastructure requirements. This represents the counterfactual cost if AL were not employed.

AL overhead ($c_{\text{AL-overhead}}$). AL overhead represents additional costs introduced by AL implementation. It includes implementation complexity, multiple training rounds, query strategy

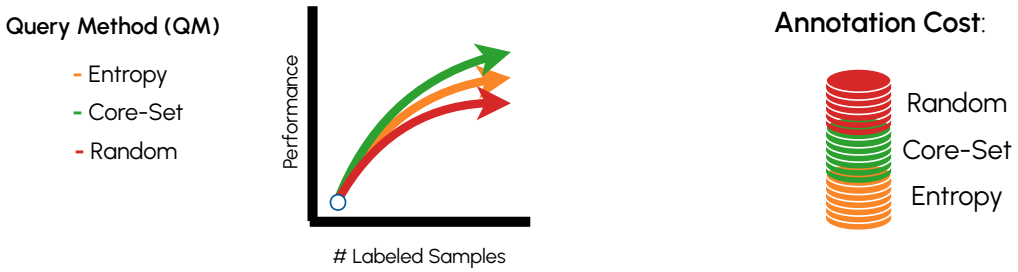


Figure 4.1: An example of the validation paradox for three potential QMs during application, where validation would triple the annotation cost of the project.

computation, additional infrastructure, and opportunity costs from delayed deployment. Critically, these costs are *certain*-they will be incurred when using AL-and measurable.

Annotation reduction (r). The annotation reduction is the reduction in labeled instances achieved by AL (i.e., the number of annotations saved). It represents the most uncertain component, influenced by AL strategy effectiveness, model characteristics, data distribution, task difficulty, and the specific instances selected. Unlike overhead costs, savings can not be evaluated during practice which we will discuss in section 4.2.

Per-instance annotation cost ($c_{\text{annotation}}$). The per-instance annotation cost represents additional costs introduced by AL implementation. It varies with task complexity, annotator expertise, annotation interfaces, and crucially, the specific instances selected. AL's selection of potentially difficult or ambiguous instances may increase per-instance costs (Settles, 2011).

Efficiency Principle. Before considering AL, projects should optimize the baseline approach through:

- Efficient annotation strategies and interfaces
- Appropriate model architectures
- Data augmentation and preprocessing
- Transfer learning from pretrained models
- Semi-supervised or self-supervised learning where applicable

These optimizations are directly measurable and can substantially reduce C_{baseline} with guaranteed benefits as they do not require additional annoations and instead make more efficient use of already present annotations. AL operates *orthogonally* to these improvements, potentially providing additional gains by means of guiding the annotation process but with uncertain outcomes. This observation partially explains why many projects stop optimization before adopting AL as conventional risk-reward calculation favors approaches with measurable, guaranteed improvements (Settles, 2011; Jaster and Kohlhase, 2025; Tomanek and Olsson, 2009).

4.2 The Active Learning Validation Paradox

Definition. When deploying AL on a new, mostly unlabeled dataset, practitioners face a methodological dilemma: determining which query strategy performs best would require labeling separate AL trajectories for each candidate method as visualized in fig. 4.1. This extensive labeling directly contradicts AL's purpose of reducing annotation effort.

This is fundamentally different from standard Machine Learning methods such as different forms of models, where fixed train/validation splits enable comparison of methods without increasing annotation cost.

However, in both standard Machine Learning and AL the computation cost increases, which however is not as problematic as both paradigms are not inherently designed to be compute efficient during training.

Implications This paradox has several consequences:

- **No direct validation:** Unlike hyperparameter tuning in standard ML, practitioners cannot validate AL strategies on their specific problem without undermining AL's value proposition.
- **Reliance on simulation studies:** Evaluation must occur in offline settings where all labels are pre-existing, creating a gap between evaluation conditions and deployment conditions.
- **Generalization requirement:** AL methods must demonstrate effectiveness across diverse settings, as practitioners cannot verify performance on their specific problem.

4.3 The Active Learning Decision Framework

From Guarantees to Expected Value. The Validation Paradox means we cannot guarantee that AL will reduce costs for any specific deployment. However, we can still provide practitioners with actionable frameworks *by shifting from individual guarantees to expected outcomes across problem classes*. This formulation acknowledges that AL may not reduce costs for every specific problem. Therefore this represents a middleground as it is weaker than formulations which postulate that "the real-life practitioner is looking for AL methods that are always better than Random sampling" (Abraham and Dreyfus-Schmidt, 2021) but stronger than "can machines learn more economically if they are allowed to ask questions?" (Settles, 2011).

The practitioner's decision reduces to comparing the expected costs for a problem with specific characteristics (fingerprint):

$$\text{Use AL if: } \mathbb{E}_{\text{fingerprint}}[C_{\text{AL}}] < \mathbb{E}_{\text{fingerprint}}[C_{\text{baseline}}] \tag{4.2}$$

Therefore it is sufficient to show that for a class of problems with similar characteristics, AL can provide cost saving. The practitioner must assess whether their problem belongs to a favorable class. This is based on the problem formulation (classification, semantic segmentation, regression), domain (natural images, molecular data, medical imaging), data structure (2D/3D images, tabular data).

Simplification of the decision In practice, estimating $\mathbb{E}_{\text{fingerprint}}[C_{\text{AL}}]$ and $\mathbb{E}_{\text{fingerprint}}[C_{\text{baseline}}]$, however, is even in a benchmarking scenario challenging. The most fundamental requirement to show the economic viability of an AL method lies in showing that the expected reduction in annotations $\mathbb{E}_{\text{fingerprint}}[r] > 0$ for a specific class of problems.

Especially in scenarios where the overall annotation cost per instance is very large, it may very well be sufficient for the decision to have strong evidence that $\mathbb{E}_{\text{fingerprint}}[r] > 0$ (see eq. (4.1)).

4.4 Requirements for Active Learning Evaluation

There is a consensus in the literature that the deployment of AL in real-world tasks seems to be less common than studies indicating positive outcomes with regard to AL and this is due to a lack of trust in current methods (Settles, 2011; Abraham and Dreyfus-Schmidt, 2021; Kottke et al., n.d.; Tomanek and Olsson, 2009). Therefore, the primary challenge AL needs to overcome for wide-spread adoption appears to be that it must *convince* the potential practitioner that employing AL will reduce the overall cost of the project.

AL evaluation is exclusively carried out in offline simulated settings where all labels for the dataset are available as it is economoically not feasible to perform rigorous evaluation in a practical scenario requiring annotations due to the validation paradox. The reason behind the lack of trust in current

evaluation protocols for classification and semantic segmentation in the 3D biomedical domain lies in the way it is simulated. We discuss this in more detail in chapter 5 and chapter 7 in the form of pitfalls.

Here, we will now define the overall four evaluation requirements (R1-R4) that are necessary to ensure convincing and trustworthy estimation of AL methods:

R1: Generalization across problem characteristics. To ensure that AL's expected benefits generalize to new, unseen problems, its evaluation must cover diverse datasets, annotation budgets, and query parameters to establish that AL benefits are not artifacts of specific settings. This requires systematic variation of:

- Dataset distributions and sizes
- Annotation budgets (small, medium, large)
- Query batch sizes
- Random seeds and initialization conditions

R2: Compatibility with orthogonal efficiency improvements. Since practitioners should and are expected to optimize the baseline approach first, AL must demonstrate benefits *in combination with* established efficiency techniques:

- Using pretrained models (self-supervised learning, transfer learning)
- Hyperparameter optimization (s.a. data augmentation, learning rate, weight decay)
- Semi-supervised learning
- Weak supervision / partial annotations

This distinguishes AL as an *additional* efficiency layer rather than a replacement for foundational optimizations and allows for more realistic assessments of observed performance gains a practitioner can expect.

R3: Meaningful baseline comparisons. AL must outperform not just naive random sampling, but computationally comparable alternatives:

- Stratified sampling
- Diversity-based sampling
- Problem-specific heuristics

These baselines establish that AL's benefits justify its additional induced cost s.a. computational and implementation overhead.

R4: Annotation cost-aware performance metrics. Evaluation must report metrics that directly relate performance to annotation cost. This means that, when the annotation effort is strongly variying across queries this must be taken into account when comparing AL methods.

Generalization vs. Specialization. Requirements R1-R4 appear to demand exhaustive evaluation across all possible settings. This is, however, not always necessary as the following design decisions can reduce the necessary evaluations:

1. **Specify scope explicitly:** Clearly define the problem class (task types, domains, scale) and specialize the evaluation based on the current state-of-the-art for label-efficient training.
2. **Test within scope rigorously:** Thoroughly evaluate within the defined scope, following R1–R4.
3. **Acknowledge limitations:** Explicitly state settings where the method has not been validated.
4. **Provide decision heuristics:** Based on evaluated characteristics, give practitioners guidelines for when methods are likely to transfer.

By setting a reasonable scope of evaluation one can trade off universal exhaustive evaluation for generalization to make conclusions of AL performance with a practically relevant scope.

Development and Held-Out Testing During development of an AL methods the risk of overfitting to the datasets it is developed on arises. While this is a common issue in ML, in the case of an AL method this is especially troublesome as the sole benefit of an AL method lies in its generalization to novel yet unseen datasets. Therefore, we advise to perform a rigorous evaluation in two phases for developing novel AL methods:

1. **Development phase:** Use subset of datasets for method development, hyperparameter selection, and ablation studies.
2. **Held-out testing:** Evaluate final method(s) on separate, previously unused datasets that simulate roll-out to new problems.

This mirrors the train-test split principle but at the dataset level, providing evidence that methods generalize beyond their development context which is also common practice in the development of general purpose methods (Isensee, Paul F. Jaeger, et al., 2021a; T. Chen et al., 2020; M. Baumgartner et al., 2021).

4.5 Related Works

We acknowledge, that we are not the first to discuss the topic of evaluating AL. Part of the requirements that we formulated in section 4.4 to increase trust in AL by means of fair and rigorous evaluation have been previously stated (Jaster and Kohlhase, 2025; Tseng et al., 2025; Kottke et al., n.d.; Abraham and Dreyfus-Schmidt, 2021; Yilin Ji et al., 2023; Munjal et al., 2022a; Mittal, Tatarchenko, et al., 2019a; Beck et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B. Chan, 2022b).

Our approach in this section differs however from previous work. By deriving these requirements from first principles and formulate them as generalizingly as possible they are easily extendible through logic for novel scenarios and moving with the state-of-the-art.

4.6 Summary

In this section we discussed AL from a meta perspective with the following high-level take aways with regard to our question:

> **Research Question:** What are general aspects necessary for the evaluation of Active Learning methods?

The evaluation of AL methods must acknowledge the following inherent properties important for practitioners:

- **Economic framework:** AL adoption is a cost-benefit decision involving certain overhead costs and uncertain annotation savings.
- **Validation Paradox:** Direct validation of AL on specific problems contradicts its purpose.
- **Decision framework:** Practitioners need expected-value assessments based on problem characteristics.

Based on these, we derive the following crucial aspects for the evaluation of AL methods:

1. **Evaluation requirements:** Convincing evaluation must demonstrate generalization, orthogonality to baseline optimizations, meaningful comparisons, and effort-aware metrics.
2. **Strategic specialization:** Thorough evaluation within a clearly defined scope allows represents a realistic approach to reduce the burden of showing generalization capabilities to all settinds.

The remainder of this thesis applies these principles to develop and evaluate AL methods for classification and semantic segmentation for 3D biomedical imaging, providing practitioners in this area with evidence-based guidance for AL adoption decisions.

Chapter 5

Principled evaluation of Active Learning for Image Classification

> Acquisition of skills requires a regular environment, an adequate opportunity to practice, and rapid and unequivocal feedback about the correctness of thoughts and actions.
>
> Daniel Kahneman

Research Question: How can we ensure generalizable and meaningful evaluation of Active Learning methods to guide practitioners in deep active classification?

The historical trajectory of deep learning in computer vision illustrates a natural progression from simpler to more complex tasks. Initial breakthroughs in object recognition and classification, exemplified by AlexNet's success on ImageNet (Alex Krizhevsky et al., 2017), established the viability of deep learning approaches before their extension to more complex scenarios such as semantic segmentation and object detection (J. Long et al., 2015). Following this paradigm, we adopt a similar progression in our investigation of AL: before addressing the challenges of 3D biomedical imaging, we first critically examine the practicality and effectiveness of AL in the classification setting. This foundational analysis in the current chapter provides essential insights that inform our subsequent exploration of AL for semantic segmentation tasks in later chapters. This chapter is based on:

- Carsten Tim Lüth, Till J. Bungert, Lukas Klein, and Paul F Jaeger (2023). "Navigating the Pitfalls of Active Learning Evaluation: A Systematic Framework for Meaningful Performance Assessment". In: *Thirty-Seventh Conference on Neural Information Processing Systems*

5.1 Problem Statement

In this chapter, we will discuss the inconsistent state of research in AL with regard to its general effectiveness creates significant challenges for practitioners who aim to annotate datasets efficiently. These are confronted with fundamental questions such as: On which tasks and datasets does AL provide a tangible benefit, and how can it be most effectively applied to a specific dataset?

This inconsistency is characterized within the AL community, where some researchers focus on developing new QMs to advance the field claiming methodological superiority and improvements over random sampling (K. Kim et al., 2021; Citovsky et al., 2021). While in contrast, others report that AL is often outperformed by alternative training paradigms to standard supervised training

(ST), most notably Semi-Supervised Learning (Semi-SL) (Mittal, Tatarchenko, et al., 2019a; Gao, Z. Zhang, Yu, Arık, et al., 2020) and Self-Supervised Learning (Self-SL) (Bengar et al., 2021), or even by carefully tuned ST baselines (Munjal et al., 2022a). Further adding to the uncertainty, several studies have observed that AL can in some settings reduce classification performance, a phenomenon known as the "cold start problem" (Gao, Z. Zhang, Yu, Arık, et al., 2020; Bengar et al., 2021; Mittal, Tatarchenko, et al., 2019a).

In the following we will critically reflect upon practices in related literature and show that the inconsistency likely arises from a lack of systematic and realistic evaluation of AL methods. AL inherently comes with specific requirements on how methods will be applied in practice. First and foremost, the stated purpose of "reducing the labeling effort on a task" implies that AL methods need to be rolled out to unseen datasets different from the labeled development data on which AL is commonly simulated. This inherent requirement of cross-task generalization needs to be reflected in method evaluation, posing a need to test methods under diverse settings to identify robust and trustworthy configurations for the subsequent "blind" application on real-life tasks (see "validation paradox", section 5.2). However, such considerations are generally neglected in AL research, as identified in our work by means of five key pitfalls in the current literature, spanning from a lack of tested AL settings and tasks to a lack of appropriate baselines (see fig. 5.1 and P1-P5 in section 5.2).

To this end, we design an evaluation framework for deep active classification that overcomes these five pitfalls and demonstrate the relevance of this contribution by means of a large-scale empirical study spanning various datasets, QMs, AL settings, and training paradigms.

Just by addressing the widespread pitfall of neglecting classifier configuration (see P4 in Figure 1b) through intrudction of a simple, light-weight protocol for high-quality configuration our results demonstrate how even our random-query baseline exceeds the originally reported performance of recent AL methods (Krishnan et al., 2021; K. Kim et al., 2021; Sharat Agarwal et al., 2020) on the widely-used CIFAR-10/100 datasets, while drastically reducing computational effort compared to the configuration protocol of Munjal et al. (2022a).

The novelty of this evaluation framework does not lie in presenting entirely novel results and insights, but in the fact that our comprehensive and systematic approach is the first to address all five key pitfalls and thus to provide *trustworthy* insights into the real-life capabilities of current AL methods.

By relating the insights based on the proposed protocol to recent studies that may be subject to flawed evaluation practices, it becomes possible to resolve existing inconsistencies in the field and provide robust guidelines for when and how to apply AL on a given task.

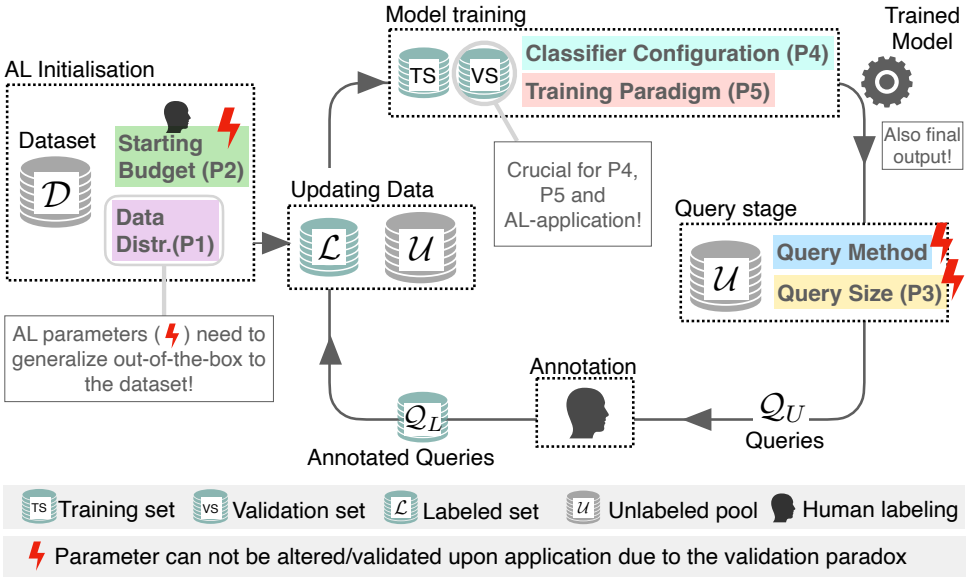
5.2 Pitfalls and Solutions for Evaluation of Active Learning for Deep Active Classification

Critical concepts in Active Learning evaluation for Classification

Evaluating an AL algorithm typically means testing how much classification performance is gained by data samples queried over several training iterations. The QM selecting those samples is considered useful if the performance gains exceed the gains of randomly queried samples. While this process is well-established, it is prone to neglecting critical concepts for evaluation and thus to over-simplification of how AL algorithms are applied in practice. Especially the *AL validation paradox* needs to be taken into account during the evalution.

Special requirements on evaluation. As the AL validation paradox impedes on-the-spot validation, it forces one to, instead, estimate how well certain QMs will perform on the given task based solely on prior knowledge. The quality of this prior knowledge depends on how extensively the respective QM has been validated prior to application, i.e. on the development data. This implies several critical requirements on evaluation which are listed in section 4.4.

The critical requirements for active learning evaluation in classification are generalization across problem characteristics (**R1**) and compatibility with orthogonal efficiency improvements (**R2**). As



(a)

Pitfall	P1	P2	P3	P4		P5	
Compared Aspect	Data	Starting	Query	Perform.	HP Optim.	Self	Semi
Related Work	Distr.	Budget	Size	Baselines	& Val Split	SL	SL
Munjal et al. (2022a)	(✔)		✔	✔	✔		
Mittal, Tatarchenko, et al. (2019a)		(✔)		(✔)			✔
Bengar et al. (2021)		✔	(✔)			✔	
Gao, Z. Zhang, Yu, Arık, et al. (2020)		✔	(✔)				✔
J. S. K. Yi et al. (2022)	✔					(✔)	
Krishnan et al. (2021)	(✔)						
K. Kim et al. (2021)	(✔)	(✔)	(✔)				
Beck et al. (2021)		(✔)	(✔)	(✔)			
Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B Chan (2022a)	✔		✔				
Y.-C. Chan et al. (2021)				✔		✔	✔
Ours	✔	✔	✔	✔	✔	✔	✔

(b)

Figure 5.1: **(a)**: The five pitfalls (P1-P5) for meaningful evaluation in the context of the Active Learning loop. Detailed information is provided in section 5.2. **(b)**: The five pitfalls are highly prevalent in the current literature (green ticks denote successful avoidance of the respective pitfall). A detailed correspondance between individual studies and pitfalls is provided in section A.1. Our study is the first to avoid all pitfalls and enable trustworthy performance assessment of AL methods. Figure is taken from Carsten Tim Lüth et al. (2023)

annotation of datasets is commonly performed based on random-batches and the annotation effort across images is usually very similar leading to the two remaining requirements meaningful baseline comparisons (**R3**) and annotation cost-aware performance metrics (**R4**) are not the focus of this chapter.

Current pitfalls of Active Learning evaluation

The current AL literature features an inconsistent landscape of evaluation protocols with many works already employing parts of our solution, but, as fig. 5.1b shows, none of them adhere to all requirements for evaluation described above. To study the current oversight of the two previously mentioned requirements **R1** and **R2**, we identify five key **pitfalls** (**P1-P5**) that need to be overcome for meaningful evaluation of QMs. For a visual overview of the pitfalls and how they integrate into the AL setting see fig. 5.1a.

P1: Lack of evaluated data distribution settings. To ensure that QMs work out-of-the-box in real-life settings, they need to be evaluated on a broad data distribution. Relevant aspects of a distribution in the AL-context go beyond the data domain and include class distribution, the relative difficulty of separation across classes, as well as a potential mismatch between the frequency and importance of classes. All of these aspects directly affect the functionality of a QM and may lead to real-life failure of AL when not considered in the evaluation. This pitfall contradicts **R1**.

Current practice: Most current work is limited to evaluating QMs on balanced datasets from one specific domain (e.g. CIFAR-10/100) and under the assumption of equal class importance. To our knowledge, testing the generalizability of a fixed QM setting to new datasets ("roll-out") has not been performed before. There are some experiments conducted on an artificially imbalanced dataset (CIFAR-LT) (Munjal et al., 2022a; Krishnan et al., 2021) suggesting good AL performance. Further, Gal, Islam, et al. (2017a) study AL on the ISIC-2016 dataset, but obtain volatile results due to the small dataset size. Atighehchian et al. (2020) study AL on the MIO-TCD dataset and reported performance improvements for underrepresented classes. In the large study Beck et al. (2021) perform experiments on five mostly balanced datasets datasets and (Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B Chan, 2022a) use 13 datasets for mostly standard AL experiments.

→ **Proposed solution:** We argue that the underrepresentation of class-imbalanced datasets in the field is one reason for current doubts regarding the general functionality of AL. Real-life settings will most likely not be class balanced providing a natural advantage of AL over random sampling. We propose to consider diverse datasets with real class imbalances as an essential part of AL evaluation and advocate for the inclusion of "roll-out" datasets, as a real-life test of selected and fixed AL settings.

P2: Lack of evaluated starting budgets. There are two reasons for why this parameter is an essential aspect of AL evaluation: 1) Upon application, the budget might be fixed and the QM is required to generalize out-of-the-box to this setting. 2) We are interested in the minimal budget at which the QM works since a too large budget implies inefficient labeling (equivalent to random queries) and a too small budget is likely to cause AL failure (cold start problem). This search needs to be performed prior to an AL application due to the validation paradox. This pitfall contradicts **R1**.

Current practice: Most recent studies evaluate AL on a single starting budget made of thousands of samples on datasets such as CIFAR-10 and CIFAR-100 (Munjal et al., 2022a; K. Kim et al., 2021; Yoo and Kweon, 2019b; J. S. K. Yi et al., 2022). Information-theoretic publications commonly use a smaller starting budget (Gal, Islam, et al., 2017a; Kirsch, van Amersfoort, et al., 2019a), but typically on even simpler datasets such as MNIST (Yann LeCun, 1998). Beck et al. (2021) compare two starting budgets on MNIST reporting no performance drop. On the other hand, some studies benchmarking AL against Semi-SL and Self-SL compare two (Bengar et al., 2021) or three (Mittal, Tatarchenko, et al., 2019a; Gao, Z. Zhang, Yu, Arık, et al., 2020) starting budgets often with the conclusion that smaller starting budgets lead to AL failure. Bengar et al. (2021) report that there

exists a relationship between the number of classes in a task and the optimal starting budget (the intuition being that class number is a proxy for task complexity).

→ **Proposed solution:** To overcome this pitfall and resolve the current contradictions, we evaluate all QMs for three different starting budgets on all datasets. We refer to these settings as the *Low-, Medium-, and High-Label Regime*. Extending on the findings of Bengar et al. (2021), adequate budget sizes are determined using heuristics based on the number of classes per task.

P3: Lack of evaluated query sizes. The number of samples queried for labelling in each AL iteration is an essential aspect of QM evaluation. This is because, upon application, this parameter might be predefined by the compute-versus-label cost ratio of the respective task (a smaller query size amounts to higher computational efforts but might enable more informed queries and thus less labeling). Since query size cannot be validated on the task at hand due to the validation paradox, the generalizability of QMs to various settings of this parameter needs to be evaluated beforehand. This pitfall contradicts **R1**.

Current practice: In current literature, there is a concerning disconnect between theoretical and practical papers regarding what constitutes a reasonable query size. Information-theoretical papers typically select the smallest query size possible and QMs such as BatchBALD (Kirsch, van Amersfoort, et al., 2019a) are specifically designed to simulate reduced query sizes (Gal, Islam, et al., 2017a; Pinsler et al., 2021). In contrast, practically-oriented papers usually employ larger query sizes (K. Kim et al., 2021; Yoo and Kweon, 2019b; Sinha et al., 2019; Mittal, Tatarchenko, et al., 2019a; Munjal et al., 2022a), but only in combination with large starting budgets (P2), where cold start problems generally do not occur. Only a few studies perform limited evaluations of varying query sizes. Beck et al. (2021) and Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B Chan (2022a) report a negligible effect of varying query sizes, but only evaluate in combination with large starting budgets (1000 samples). In line with this, Munjal et al. (2022a) conclude that the choice of query size does not matter, but only compared two large values (2500 versus 5000 samples) on a fixed large starting budget (5000 samples). Atighehchian et al. (2020) come to a similar conclusion, but also only considered a relatively large starting budget (500) for ImageNet-pretrained models on CIFAR-10, where, again, no cold start problem occurs. Bengar et al. (2021) employ varying query sizes without further analysis of the parameter.

→ **Proposed solution:** To overcome this pitfall and reliably study the effect of query sizes also in Low-label, i.e. high-risk, settings, we evaluate all QMs for three different query sizes in combination with varying starting budgets on all datasets (i.e. as part of the Low-, Medium-, and High-Label Regimes). For a specific focus on the effect of query size in the Low-label settings, we perform an additional ablation with varying query sizes on a small fixed starting budget.

P4: Neglection of the classifier configuration. As stated in section 5.2, when aiming to draw conclusions about the performance or usefulness of a QM, it is critical that this evaluation be based on well-configured classifiers. Otherwise, performance gains might be attributed to AL that could have been achieved by simple hyperparameter (HP) modifications. Separating a validation split from the training data is a crucial requirement for sound HP tuning. This pitfall contradicts **R2**.

Current practice: Most studies in AL literature do not report how HPs are obtained and do not mention the use of validation splits (Krishnan et al., 2021; Yoo and Kweon, 2019b; Sinha et al., 2019; K. Kim et al., 2021; Mittal, Tatarchenko, et al., 2019a). Typically, reported settings are copied from fully labeled data scenarios. In some cases, even the proposed QMs feature delicate HPs without reporting how they were optimized raising the question of whether these settings generalize to new data (Sinha et al., 2019; K. Kim et al., 2021; Yoo and Kweon, 2019b). Munjal et al. (2022a) demonstrate how adequate HP tuning on a validation set allows a random query baseline to outperform current QMs under their originally proposed HP settings. However, they run a full grid search for every QM and AL training iteration, which might not be feasible in practice.

→ **Proposed solution:** To overcome this pitfall and enable meaningful performance assessment of AL methods, we define a validation dataset of a size deducted heuristically from the starting budget. Based on this data, a small selection of HPs (learning rate, weight decay and data augmentations

(Cubuk et al., 2020)) is tuned only once per AL experiment while training on the starting budget. The limited search space and discarding of multiple tuning iterations result in a lightweight and practically feasible protocol for classifier configuration.

P5: Neglection of alternative training paradigms. Analogously to arguments made in P4, meaningful evaluation of AL requires comparison against alternative approaches that address the same problem. Specifically, the training paradigms Self-SL (T. Chen et al., 2020; K. He, H. Fan, et al., 2020) and Semi-SL (Sohn, Berthelot, Carlini, et al., 2020; D.-H. Lee et al., 2013) have shown strong potential to make efficient use of an unlabeled data pool in a classification task thus alleviating the labeling burden. Additionally to benchmarking AL against Self-SL and Semi-SL, the question arises of whether AL can yield performance gains when combined with these paradigms. This pitfall contradicts **R2**.

Current practice: While most AL studies do not consider Self-SL and Semi-SL, there are a few recent exceptions: Bengar et al. (2021) benchmark AL in combination with Self-SL and conclude that AL only yields gains under sufficiently high starting budgets. However, these results suffer from inadequate classifier configuration (P4). J. S. K. Yi et al. (2022) propose a QM in combination with Self-SL, but the employed Self-SL strategy is limited by compatibility with the proposed QM. Further, Gao, Z. Zhang, Yu, Arık, et al. (2020) combine Semi-SL with AL and report superior performance compared to ST for CIFAR-10/100 and ImageNet (Russakovsky et al., 2015b), i.e. the datasets on which Semi-SL methods have been developed. Similarly, Mittal, Tatarchenko, et al. (2019a) evaluate the combination of Semi-SL and AL on CIFAR-10/100, reporting strong improvements compared to ST and find that AL decreases performance for small starting budgets.

→ **Proposed solution:** To overcome this pitfall and resolve current inconsistencies, we benchmark all QMs against both Self-SL and Semi-SL, and evaluate the combination of AL with these paradigms. Crucially, we are the first to study these relations as part of a reliable evaluation, i.e. while avoiding all other key pitfalls (P1-P4).

5.3 Empirical Study

5.3.1 Setup

This section describes the design of our empirical study in light of the proposed improvements for AL evaluation (detailed experimental settings can be found in section A.3). We first address P1 by extending our evaluation to 5 different datasets, containing different label distributions. Specifically, these datasets include CIFAR-10, CIFAR-100, CIFAR-10 LT, ISIC-2019 and MIO-TCD, where the first three are developmental datasets and the latter two are used exclusively for the proposed roll-out evaluation. Further, we address P2 and P3 by defining three different *Label Regimes* which we refer to as "Low-Label", "Medium-Label" and "High-Label" Regimes. Starting budgets and query sizes are both set to $5 \times C$, $25 \times C$ and $100 \times C$ for the three Label Regimes, where C denotes the number of classes.[1] To address P4, we configure our classifiers for all three Label Regimes based on a validation set five times the size of the starting budget. Further, addressing P4 and P5 we use a ResNet-18 (K. He, X. Zhang, et al., 2016) as the backbone in all experiments and optimize the essential HPs for each respective training paradigm. At last, we address P5 by comparing randomly initialized models (standard training) against Self-SL pre-trained models and Semi-SL models.

Compared query methods. We focus exclusively on QMs which do not alter the classifier and require no configuration of additional HPs. In the experiments the following methods are used with more detailed explanations in section 3.1.1.

Random: The baseline all QMs are compared against which randomly draws samples from the pool $\mathcal{U}$.

Core-Set: This explorative QM aims to find the core-set of a convolutional neural network (Sener

[1] These deviate for CIFAR-100 $5 \times C$ (Low-), $10 \times C$ (Medium-), $50 \times C$ (High-Label) due to a smaller ratio of dataset size to number of classes.

and Savarese, 2018b) by means of a K-Center Greedy approximation on the representations used by the classification head.

Entropy: This uncertainty-based QM selects the samples with the highest entropy across predicted class scores (Settles, 2009).

BALD: This uncertainty-based QM uses the mutual information between the class label and the model parameters with regard to each sample for top-k selection (Houlsby et al., 2011), it was introduced with dropout for deep bayesian active learning(Gal, Islam, et al., 2017a).[2]

BADGE: This QM performs a clustering based on per-sample gradient vectors obtained via proxy labels. This enables a selection that is both diverse and guided by uncertainty (J. T. Ash et al., 2020).

Development Datasets. We evaluate our approach on three well-established image classification benchmarks that differ in both the number and distribution of semantic classes:

CIFAR-10 (A. Krizhevsky, 2009) consists of small natural images covering 10 broad object categories such as animals (e.g., birds, cats, dogs) and vehicles (e.g., airplanes, trucks) with an image size of 32×32. All classes are uniformly represented and it has an official training and test split of 50,000 and 10,000 images respectively.

CIFAR-100 (A. Krizhevsky, 2009) follows the same image format but expands the label space to 100 fine-grained classes, making the task significantly more challenging while maintaining a uniform class distribution. It has an official training and test split of 50,000 and 10,000 images respectively.

CIFAR-10-LT (Cao et al., 2019) is a long-tailed variant of CIFAR-10, where the training set follows an exponential class-imbalance. We use an imbalance factor of $\rho = 50$ following (Krishnan et al., 2021), creating a realistic scenario where some classes have far fewer training examples.

For all three datasets, we report classification accuracy on the original, class-balanced test sets.

Roll-out Datasets. As roll-out datasets to further test generalization capabilities, we selected two datasets featuring class imbalances which are also more likely to feature realistic label noise:

ISIC-2019(Tschandl et al., 2018; Codella et al., 2018; Combalia et al., 2019) is a dataset with dermoscopic images for skin-lesion classification featuring 7 classes with strong variations in class frequency. The overall size of the dataset is 25,331 images. We use a custom split for training (60%), validation (15%) and testing (25%) and resize the images to 224×224.

MIO-TCD(Z. Luo et al., 2018) is a large-scale traffic surveillance dataset with 11 object categories captured from roadside cameras and also features strong class-imbalances. It covers road users and vehicles of varying size and appearance and has a size of 519,164 images. We again use a custom split for training (60%), validation (15%) and testing (25%) and resize the images to 224×224

For both MIO-TCD and ISIC-2019, we use balanced accuracy as the primary performance measure (section A.3).

Active learning setup. We report performance measures for each dataset on identical test splits based on three experiments using different seeded models and different train and validation splits to reduce the possible influence of these parameters on our results. The entire annotation budget for each Label Regime is divided into to parts where the first $\frac{1}{10}$ corresponds to the starting budget while the rest is queried with the QMs where the query size is also equal to $\frac{1}{10}$ ($\frac{3}{10}$ for Semi-SL). Further, we train the models from scratch on every training step to avoid correlated queries (Kirsch, van Amersfoort, et al., 2019a).

Training paradigms. Randomly initialized and supervised-trained models are referred to as standard trained (ST) models. Further, we use the popular contrastive SimCLR (T. Chen et al., 2020) training as a basis for Self-SL pre-training. These models are fine-tuned and are referred to as Self-SL models. Self-SL models have a two-layer MLP as a classification head to make better use of the representations (ablation in section A.4).

[2]This QM requires dropout to enable multiple predictions to treat it as a bayesian QM

For ST and Self-SL models, we obtain bayesian models by adding dropout to the classification head after the convolutions following (Gal, Islam, et al., 2017a).

As a Semi-SL method, we use FixMatch (Sohn, Berthelot, Carlini, et al., 2020) which combines the principles of consistency regularization and uncertainty reduction. Due to the long training times (factor 80) compared to ST training, we only run experiments in the Low- and Medium-Label Regime while increasing the query size by a factor of three to reduce training costs.

For imbalanced datasets, we use oversampling for ST and Self-SL pre-trained models Buda et al. (2018) and use weighted cross-entropy-loss and distribution alignment for FixMatch. Model selection for ST and Self-SL models is based on the best validation set epoch, while for Semi-SL models the final checkpoint is used.

Hyperparameter selection. For each Label Regime, we select HPs on the corresponding validation set before starting the AL loop with optimizer, scheduler and number of epochs constant across all experiments with only the batch size selected based on the image size.

For Self-SL and ST models only the learning rate, weight decay and data augmentation are optimized based on the validation set whereas for Semi-SL only learning rate and weight decay are selected as for FixMatch the data augmentation is inherently required for its methodology.

The optimization is performed on a grid with with logarithmic scaling for the learning rate and weight decay. The data augmentations used are standard augmentations and Randaugment which uses stronger augmentations acting as a regularization (Cubuk et al., 2020).

Model selection for ST and Self-SL models is based on the best validation set epoch, while for Semi-SL models the final checkpoint is used.

Low-Label query size ablation. To investigate the effect of query size in the Low-Label Regime, we conduct an ablation with Self-SL pre-trained models on CIFAR-100 and ISIC-2019. For CIFAR-100 the query sizes are 50, 500 and 2000, while for ISIC-2019 they are 10, 40 and 160.

5.3.2 Results

The results of our empirical study are presented in fig. 5.2. A detailed breakdown of results for individual datasets, as well as analyses based on the pairwise penalty matrix (J. T. Ash et al., 2020) and the area under the budget curve (Zhan, H. Liu, et al., 2021), are provided in section A.5. In the following, we discuss the main findings in relation to the five identified evaluation pitfalls (P1-P5). Our findings highlight the importance of the proposed protocol for realistic evaluation and demonstrate its potential to produce trustworthy insights into when and how AL is effective. Unless stated otherwise, all references in this chapter refer to AL studies.

P1 Data distribution. The proposed evaluation across a diverse set of dataset distributions, including dedicated roll-out datasets, proved essential for a realistic assessment of QMs as well as alternative training strategies. A central insight is that the class distribution of a dataset is a strong predictor of the potential performance gains of AL. We observe that these gains are generally larger on imbalanced datasets and arise consistently even for ST models trained with a small initial budget, a setting that is typically susceptible to cold start problems. This finding aligns with previous observations (Krishnan et al., 2021; J. S. K. Yi et al., 2022; K. Kim et al., 2021).

Our results further emphasize the critical role of the roll-out datasets. For instance, we find sub-random performance of BALD (with Self-SL) and Entropy (with ST) on MIO-TCD. Such worst-case failures of AL—where compute and labeling effort increase without performance gains—could not have been anticipated from development data on which all AL parameters were optimized. A similar pattern is evident in the limited generalizability of Semi-SL, whose relative performance compared with Self-SL and ST decreases progressively with increasing dataset complexity, from CIFAR-10 to CIFAR-100 and CIFAR-10 LT, and further to the roll-out datasets MIO-TCD and ISIC-2019.

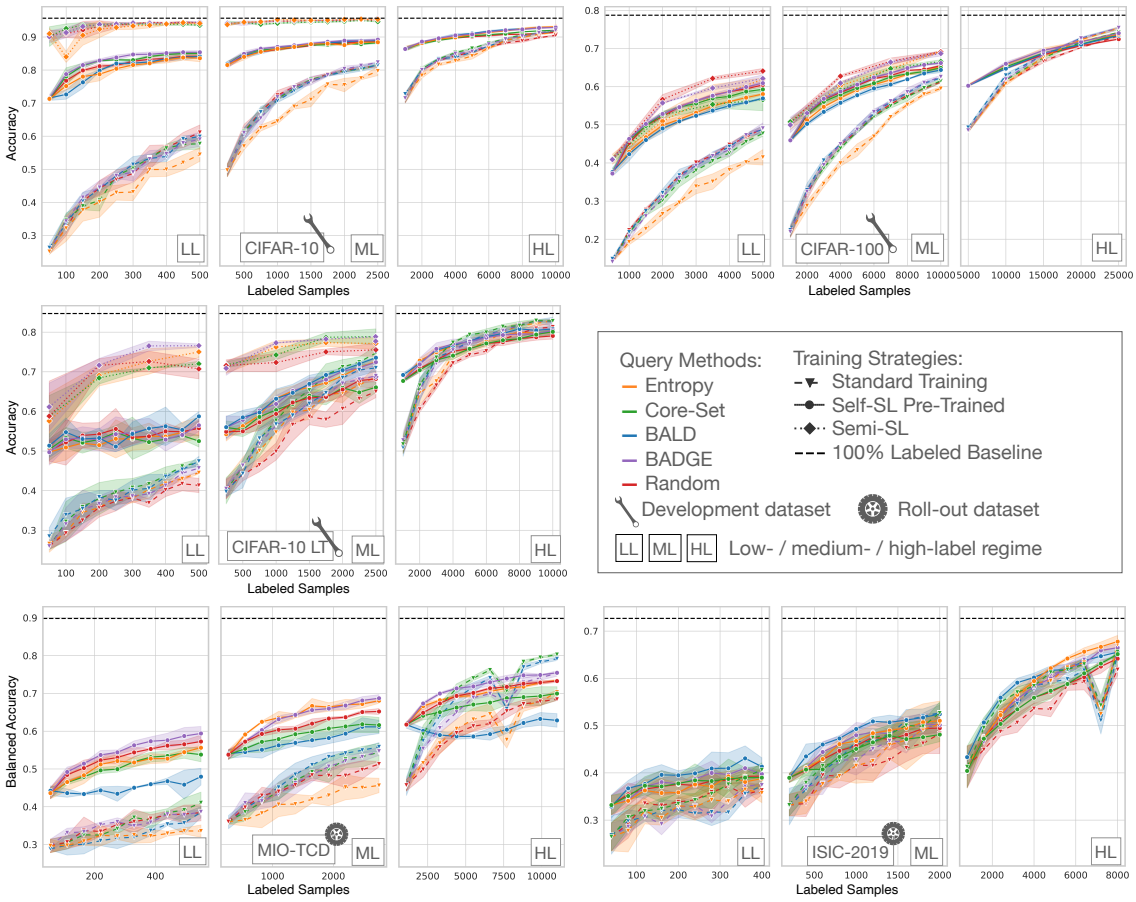


Figure 5.2: Results obtained with our proposed evaluation protocol over five different datasets and the three Label Regimes. These experiments are to our knowledge the largest conducted study for AL and reveal insights along the lines of the five key parameters as discussed in section 5.2. The performance dip on MIO-TCD and ISIC-2019 is discussed in section 5.3.2. Figure is taken from Carsten Tim Lüth et al. (2023).

P2 Starting budget. Our comprehensive study of different starting budgets across all datasets shows that AL methods are more robust to small starting budgets than previously reported (Bengar et al., 2021; Gao, Z. Zhang, Yu, Arık, et al., 2020; Mittal, Tatarchenko, et al., 2019a). With the exception of Entropy, we did not observe cold start problems for any QM, even when combined with ST models that are typically considered prone to this issue. This robustness is likely enabled by our carefully configured classifiers (P4) and the heuristically adapted query sizes (P3). The finding has notable practical implications: it suggests that AL can be introduced earlier in the annotation process, thereby further reducing labeling costs, particularly when using BADGE, which consistently performed as well as or better than Random.

For Self-SL models, AL likewise performs well with small starting budgets, except on CIFAR-100 (for BALD and Entropy) and on MIO-TCD (for BALD, Core-Set, and Entropy).

P3 Query size. Our evaluation of the query size empirically confirms its importance for (1) overall AL performance and (2) mitigating the cold start problem. Nevertheless, several findings highlight that the precise relationship between query size and performance remains an open research question. For example, we observe a pronounced cold start problem for BALD with Self-SL on CIFAR-100, where accuracy drops to about 50% at 2 k labeled samples for query sizes of 500 (Low-Label Regime) and 1 k (Medium-Label Regime). In contrast, in the High-Label Regime (5 k budget and query size of 5 k), ST and Self-SL models—achieving similar accuracies of roughly 50% and 60% benefit from BALD. Since cold start issues are typically associated with large query sizes, this result appears counter-intuitive, although it has been reported previously without further investigation (Bengar et al., 2021; Gao, Z. Zhang, Yu, Arık, et al., 2020; Mittal, Tatarchenko, et al.,

2019a).

To better understand how QMs interact with query size in the Low-Label Regime, we conducted a dedicated experiment series for Self-SL training on CIFAR-100 and ISIC-2019 (see fig. 5.3 and section A.5.2). For BALD, we observe a clear improvement on CIFAR-100 when using even smaller query sizes: performance rises from sub-Random to Random levels, while ISIC-2019 shows no degradation. We therefore conclude that small query sizes represent an effective countermeasure against the cold start problem for BALD—contrary to the findings of (Munjal et al., 2022a; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B Chan, 2022a) and not currently considered in existing solutions (Mittal, Tatarchenko, et al., 2019a; Gao, Z. Zhang, Yu, Arık, et al., 2020; Bengar et al., 2021). This observation may also help explain the discrepancy between empirical results and more theoretical works that recommend using the smallest query size possible (Gal, Islam, et al., 2017a; Kirsch, Farquhar, et al., 2022).

Importantly, while smaller query sizes appear to stabilize BALD, they have no notable effect on the other QMs. Moreover, our ablation shows that even under low-label conditions, BADGE remains the most reliable of all compared QMs and exhibits no sub-Random performance. This extends existing reports of BADGE's robustness in higher Label Regimes (Beck et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B Chan, 2022a).

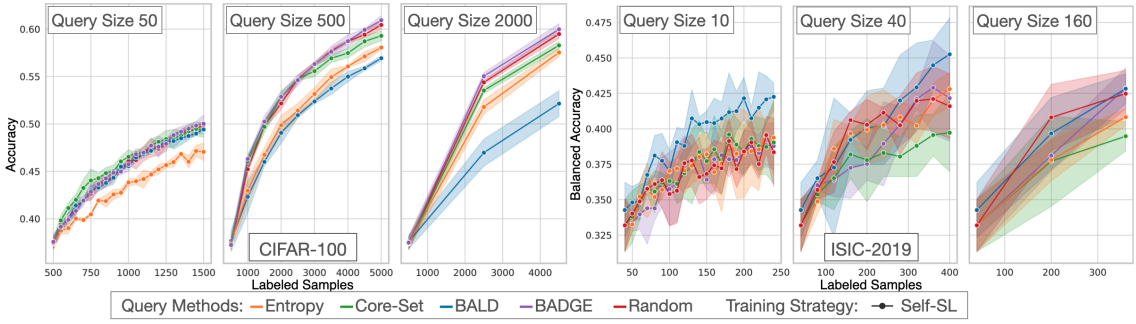


Figure 5.3: Low-Label query size ablation on CIFAR-100 and ISIC-2019. On CIFAR-100, reducing the query size resolves the observed failure mode of BALD. However, no improvement is observed on ISIC-2019, presumably because BALD already shows the best performance. Further, BADGE performs consistently well across all query sizes without failure modes, revealing its robustness also for low-label settings. Figure is taken from Carsten Tim Lüth et al. (2023).

P4 Classifer configuration. Our results demonstrate that method configuration on an appropriately sized validation set is crucial for realistic evaluation in AL. For example, our configuration improves classifier performance to a degree comparable with increasing the number of labeled training samples by a factor of approximately five; specifically, our ST model achieves around 44% accuracy when trained on 200 samples, which is similar to the accuracy reported by Bengar et al. (2021) for models trained on 1k samples.

The effectiveness of our proposed lightweight hyperparameter (HP) selection, applied only to three parameters on the starting budget (section 5.3.1), is further evidenced by the fact that all our ST models substantially outperform comparable models in prior studies (Yoo and Kweon, 2019b; Krishnan et al., 2021; K. Kim et al., 2021; Mittal, Tatarchenko, et al., 2019a; Gao, Z. Zhang, Yu, Arık, et al., 2020; Bengar et al., 2021; J. S. K. Yi et al., 2022), where HP optimization is generally neglected. Details of this comparison are provided in section A.7. This observation raises the question of whether previously reported AL advantages could in part be attributed to suboptimal classifier configurations. Notably, our models also outperform more extensively configured models reported by Munjal et al. (2022a).

We conclude that constraining the HP search space enables practical optimization without sacrificing performance and ensures that reported gains from AL are not overstated. The importance of optimizing HPs on the starting budget for each new dataset is further supported by the fact that the resulting configurations vary across datasets.

P5 Alternative training paradigms. Based on our benchmarking of AL in the context of both Self-SL and Semi-SL, we observe that Self-SL generally yields performance improvements across all experiments, whereas Semi-SL leads to substantial gains only on simpler datasets such as CIFAR-10 and CIFAR-100, which are typically the focus of Semi-SL method development. In general, models trained with either Self-SL or Semi-SL benefit less from AL (relative to random querying) compared to ST models.

A key observation is that Self-SL models converge approximately 2.5 times faster than ST models, while Semi-SL models require roughly 80 times more training time and often provide only marginal improvements over Self-SL or ST. Since AL involves multiple iterative training cycles, the computational cost of combining AL with Semi-SL becomes prohibitive in most practical scenarios. Moreover, our Semi-SL models based on FixMatch do not generalize well to more complex datasets, which contrasts with previous claims that Semi-SL can render AL redundant (Gao, Z. Zhang, Yu, Arık, et al., 2020; Mittal, Tatarchenko, et al., 2019a). Interestingly, the datasets where Semi-SL fails to provide benefits are precisely those where AL shows the most advantage.

This discrepancy with prior literature underscores the importance of our proposed evaluation protocol, which explicitly tests a method's generalizability to unseen datasets. This is particularly critical for Semi-SL, whose performance is known to be unstable on noisy or class-imbalanced datasets, as highlighted in prior studies (Oliver et al., 2019; Boushehri et al., 2020; Zenk et al., 2022; L.-Z. Guo, Z.-Y. Zhang, et al., 2020; L.-Z. Guo, Z. Zhou, et al., 2022; J. Kim et al., 2020).

Limitations.

We propose a lightweight and practically feasible strategy for hyperparameter (HP) optimization and made additional design choices, such as using a ResNet-18 classifier. Consequently, we cannot guarantee that our configurations are fully optimal for all compared training paradigms, and a critical discussion is warranted.

1. The combination of ResNet-18 with shortened training schedules may limit the performance of Semi-SL more than other training paradigms. This compromise is necessary to manage the substantial computational cost of Semi-SL, which can be approximately 400 times higher than ST.

2. The validation set size of 5× starting budget size (i.e. training set) could be considered as larger than practically desirable, where most data would be used for training. This design decision follows the study of Oliver et al. (2019), showing that an adequately sized validation set is necessary for proper HP selection (especially for Semi-SL).

3. We observe a performance dip of ST models on MIO-TCD and ISIC-2019 at ∼7k samples, which we attribute to our HP selection scheme. This indicates that HPs might need to be re-selected occasionally at certain training iterations. However, such cases are immediately detected in practice allowing for correction where necessary by re-optimizing the HPs (see section A.8.1).

A more extensive discussion of limitations can be found in section A.8.

5.4 Discussion

Our experiments provide strong empirical evidence that current evaluation protocols are insufficient to answer the central question faced by any potential AL practitioner: *Will Active Learning provide value for my specific dataset?* Addressing this question requires estimating whether an AL algorithm will yield performance gains over random querying and whether the expected reduction in labeling cost justifies the additional computational and engineering effort associated with AL deployment.

Research Question: How can we ensure generalizable and meaningful evaluation of Active Learning methods to guide practitioners in deep active classification?

Our proposed protocol for realistic evaluation constitutes a crucial step toward enabling such informed decisions. This is achieved by explicitly assessing the generalizability of AL to new settings through simulation of real-world conditions—a perspective operationalized through five key pitfalls in the current AL literature concerning data distribution, starting budget, query size, classifier configuration, and alternative training paradigms (see section 5.2). While this thorough evaluation increases computational cost during method development, it is expected to reduce overall costs in the long term by supporting the selection of robust AL methods, thereby effectively lowering annotation effort upon practical deployment. Our entire evaluation framework is accessible at `https://github.com/IML-DKFZ/realistic-al`.

Main empirical insights revealed by our study.

- **Classifier configuration is critical:** Assessment of AL methods in the literature is substantially limited by suboptimal classifier configurations; meaningful evaluation can be restored using our protocol for lightweight hyperparameter tuning on the starting budget.
- **Class imbalance favors AL:** AL generally provides substantial performance gains in class-imbalanced settings.
- **BADGE emerges as the most robust method:** BADGE is the best-performing query method across a realistic range of datasets, starting budgets, and query sizes, exhibiting nearly no failure modes (i.e., no sub-random performance).
- **Self-supervised learning amplifies AL benefits:** Combining AL with Self-SL considerably improves performance, reduces training time, and stabilizes optimization, particularly in Low-Label Regimes.
- **Semi-supervised methods show limited generalization:** Semi-SL methods based on FixMatch perform well on datasets for which they were developed, but struggle to generalize to more realistic scenarios, such as class-imbalanced datasets. Combining Semi-SL with AL is further limited by extensive training times.
- **Query size optimization matters:** BALD with Self-SL pre-trained models benefits from smaller query sizes in low starting budget settings, potentially mitigating the cold start problem.

Take-aways for developing and proposing new AL algorithms. AL methods should be evaluated for generalizability on roll-out datasets and accompanied by clear guidelines for real-world deployment, specifying how all design choices can be adapted to new settings. Since the expected benefit of AL increases when application costs are lower, there is substantial potential for widespread real-world adoption by addressing two prevalent cost factors: (1) *engineering costs*, which can be mitigated by providing user-friendly AL tools, and (2) *computational costs*, which can be reduced by incorporating methods that shorten training time within the AL loop.

Take-aways for the cost-benefit analysis of deploying AL:

1. As BADGE is identified as a robust query method exhibiting no sub-random performance across all tested settings, the potential risks of AL are minimized, allowing deployment decisions to be guided by a structured cost-benefit analysis.
2. The expected benefit of AL is highest in settings where there is a mismatch between the task-specific importance of individual classes and their frequency in the dataset (e.g., class-imbalanced datasets requiring balanced classifiers).
3. The expected benefit further increases with the labeling cost that AL can reduce. AL is therefore most likely to yield a net benefit in scenarios with high labeling cost and relatively low computational and engineering costs.

Future research. Foundation models such as CLIP (Radford, J. W. Kim, et al., 2021) present promising opportunities for future AL research. These models have been shown to reach strong performance in low-label settings through fine-tuning or knowledge extraction. Additionally, their rapid re-fine-tuning during AL iterations may enable real-time interactive labeling workflows. Further investigation into the interplay between foundation models and AL strategies could provide valuable insights for practitioners deploying AL in resource-constrained scenarios.

Chapter 6

Principled evaluation of Uncertainty Estimation for Semantic Segmentation

> Do not be afraid to skip equations (I do this frequently myself).
>
> Roger Penrose

Research Question: How can systematic analysis of uncertainty estimation for semantic segmentation inform the design of Active Learning query methods?

Uncertatiny estimation represents a core building block of AL, this chapter primarily focuses on improving the understanding of uncertainty methods for semantic segmentation, while extending the analysis with a more thorough examination of the implications and actionable insights for AL query method design.

This chapter is based on:

- Kim-Celine Kahl, Carsten T. Lüth, Maximilian Zenk, Klaus Maier-Hein, and Paul F Jaeger (2024b). "ValUES: A Framework for Systematic Validation of Uncertainty Estimation in Semantic Segmentation". In: *The Twelfth International Conference on Learning Representations*

This work with joint first authorship presents a framework for the systematic evaluation of uncertainty estimation methods in semantic segmentation. It is the result of a collaborative effort: theoretical foundations were developed by Carsten Lüth; the literature review was conducted jointly by Carsten Lüth and Kim-Celine Kahl; the experimental setup and implementation were designed and executed by Kim-Celine Kahl; and the analysis and interpretation of results were performed collaboratively by both authors. Given this joint development, all results are presented without individual attribution, as a comprehensive view of the complete findings is essential for drawing coherent conclusions regarding uncertainty estimation for semantic segmentation.

6.1 Problem Statement

Reliably deploying image segmentation systems in practice requires accurate estimation and quantification of the uncertainty associated with their predictions. It is a core building block for downstream task such as Active Learning (AL), failure detection and ambiguity modeling. Despite substantial research on uncertainty methods for segmentation in recent years, their effective use remains hampered by a persistent gap between theoretical development and application in relevant downstream tasks.

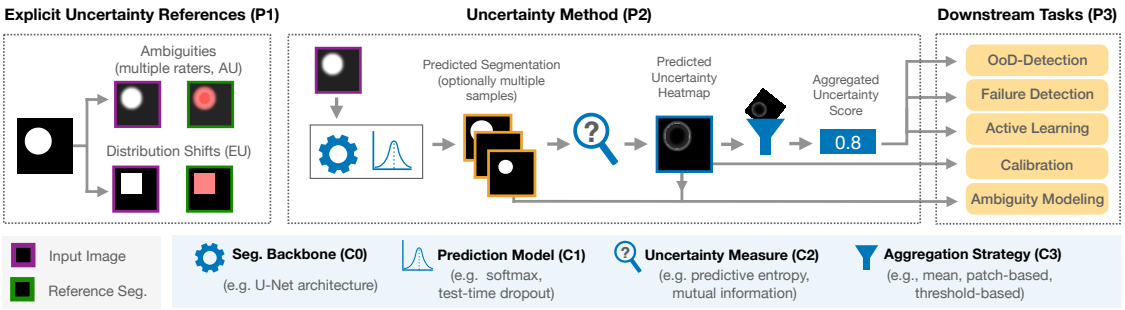


Figure 6.1: **Framework for systematic validation of uncertainty methods in segmentation**. With our framework, we aim to overcome pitfalls in the current validation of uncertainty methods for semantic segmentation by means of our solutions of the three pitfalls (P1-P3) hindering a systematic validation: We explicitly control for aleatoric and epistemic uncertainty in the data and references (P1). We define and validate four individual components C0-C3 of uncertainty methods (P2): First, one or multiple segmentation outputs are generated by the segmentation backbone (C0) and the prediction model (C1). Next, an uncertainty measure is applied (C2) producing an uncertainty heatmap, which can be aggregated using an aggregation strategy (C3). Finally, the real-world capabilities of methods need to be validated on various downstream tasks (P3). Figure is taken from Kahl, Carsten T. Lüth, et al. (2024b).

A striking example of this gap is the frequent claim that specific methods capture either data-related (aleatoric, AU) or model-related (epistemic, EU) uncertainty. Especially for AL the ability of an uncertainty method to reliably separate these two sources of uncertainty on real-world datasets would allow to actively focus on epistemic for the query step. Yet, explicit empirical validation of such claims is commonly absent. For instance, two highly cited studies assert—without theoretical or experimental support—that test-time data augmentation (TTA) enhances a model's ability to capture aleatoric uncertainty (G. Wang et al., 2019; Ayhan and Berens, 2018). Conversely, a subsequent study posits the opposite, namely that TTA models epistemic uncertainty, but likewise provides no explicit validation (S. Hu et al., 2019). This inconsistency highlights the need to verify whether the supposed separation between aleatoric uncertainty and epistemic uncertainty is actually feasible, and to assess whether such a separation offers tangible benefits for downstream applications.

Another underexplored issue concerns the practical components of uncertainty methods. For example, aggregation of pixel-level uncertainty estimates to the image level is critical in many tasks, yet often neglected. Failure detection, for instance, is frequently evaluated only at the pixel level (G. Zhang et al., 2022; Mehta et al., 2020), or simplistic aggregation strategies are employed (Gonzalez et al., 2021; Czolbe et al., 2021), limiting the reliability of conclusions. Moreover, proposed methods are rarely validated across a broad range of downstream tasks, making it difficult—and costly—for practitioners to identify the most suitable uncertainty method for their specific use case.

In this chapter, we address these shortcomings by presenting a framework for standardized and systematic validation of uncertainty methods in segmentation (see fig. 6.1). The framework provides (1) a controlled environment for analyzing both data ambiguities and distribution shifts, (2) systematic ablations of all relevant method components, and (3) dedicated testbeds for five key uncertainty applications: Out-of-Distribution detection (**OoD-D**), AL, failure detection (**FD**), calibration (**CALIB**), and ambiguity modeling (**AM**). Through an exemplary empirical study, we demonstrate the effectiveness of the proposed framework, clarify open questions and inconsistencies in the field, and distill a set of concrete, hands-on recommendations for practitioners.

6.2 Components of Uncertainty Estimation in Semantic Segmentation

We start by establishing a common terminology defining the componentes of an uncertainty method to effectively discuss pitfalls and challenges in uncertainty estimation for segmentation.

C0 - Segmentation Backbone. The segmentation backbone represents the fundamental architectural building block upon which uncertainty estimation methods are constructed—for example, a U-Net (Ronneberger et al., 2015). Given the availability of well-established architectures, this component typically remains fixed across most uncertainty studies.

C1 - Prediction Model. The prediction model extends the segmentation backbone to produce final class-score predictions for each pixel. Depending on its design, the prediction model may generate either a single set of class scores (deterministic) or multiple sets (sampling-based) for a given input image. This component may incorporate dedicated training or inference schemes, such as ensemble training or test-time dropout (TTD). Examples include deterministic softmax models, Bayesian approaches, and probabilistic models such as stochastic segmentation networks (SSNs) (Monteiro et al., 2020).

C2 - Uncertainty Measure. The uncertainty measure transforms predicted class scores into pixel-wise uncertainty scores, typically visualized as an uncertainty heatmap. Common examples include expected entropy and mutual information.

C3 - Aggregation Strategy. Unique to semantic segmentation (as opposed to image classification), the aggregation strategy converts pixel-level uncertainty heatmaps into scalar values at the granularity required by the downstream task—for instance, at the patch level or image level. Simple aggregation approaches include computing the mean or sum of all pixel-level uncertainties.

6.3 Pitfalls and Solutions for a Systematic Validation of Uncertainty Methods

Our goal is to bridge the gap between theoretical development and practical application of uncertainty methods in semantic segmentation. To this end, we identify three key pitfalls (P1–P3) that are widespread in current evaluation protocols and hinder holistic assessment of uncertainty methods for semantic segmentation. For each pitfall, we formulate concrete solutions that guide the design of our empirical study. Through these solutions, we aim to clarify how uncertainty methods behave in real-world applications, thereby enabling the safe and reliable deployment of segmentation systems in practice.

P1: Missing evaluation of uncertainty methods regarding the separability of aleatoric and epistemic uncertainty Theoretical studies often claim that a given uncertainty method captures either epistemic or aleatoric uncertainty. A clear separation of these two types of uncertainty would be highly beneficial: for instance, AL could focus on samples dominated by epistemic uncertainty, while accurate modeling of aleatoric uncertainty could substantially improve calibration on i.i.d. datasets. However, as we will show below, the current literature lacks a systematic validation of such uncertainty modeling, leaving fundamental questions unresolved: Can aleatoric and epistemic uncertainty be separated in practice? And to what extent would downstream applications benefit from such a separation?

Current practice: Several aleatoric uncertainty studies rely on test sets with only a single rater (G. Wang et al., 2019; Whitbread and Jenkinson, 2022; A. Kendall and Gal, 2017b), while many epistemic studies omit distribution shifts entirely during evaluation. This practice leaves it unclear in the context of Calibration or Failure Detection whether failure cases stem from inherent aleatoric uncertainty (ambiguity) or epistemic uncertainty (lack of similar data points) in a sample. Further, current studies commonly do not explitily evalute aleatoric uncertainty and epistemic uncertainty separation and instead report segmentation performance (G. Zhang et al., 2022), calibration (G. Wang et al., 2019; Postels, Ferroni, et al., 2019), failure detection (G. Zhang et al., 2022; Mukhoti, van Amersfoort, et al., 2021; Mobiny et al., 2021), or base claims on visual inspection Mukhoti, van Amersfoort, et al., 2021; G. Wang et al., 2019; Whitbread and Jenkinson, 2022; Mobiny et al., 2021. However, it compares only raw epistemic uncertainty scores across datasets instead of assessing

separation power with AUROC and uses predictive entropy as an epistemic uncertainty measure, thereby contradicting eq. (2.10). As a consequence of these shortcoming in the evaluation, confusion and contradictions arise, such as the fact that different studies claim TTA to either specifically capture epistemic uncertainty (S. Hu et al., 2019) or aleatoric uncertainty (Ayhan and Berens, 2018; G. Wang et al., 2019) without providing quantitative evidence for their claim.

→ **Proposed solution**: To overcome this pitfall and reliably study the ability to separate aleatoric uncertainty from epistemic uncertainty, we validate this claimed behavior 1) for aleatoric uncertainty a test set with references from multiple raters that reflect the ambiguities in the data and a metric that explicitly assesses the capturing of these ambiguities such as the normalized cross-correlation (NCC) (S. Hu et al., 2019)), and 2) for epistemic uncertainty a test set featuring samples with explicit distribution shift (i.e. induced epistemic uncertainty) and a metric that explicitly assesses whether an epistemic uncertainty-measure can separate these cases, such as the Area Under the Receiver Operating Characteristic Curve (AUROC). Further, we design a specific study to answer the open questions regarding the separation of epistemic uncertainty and aleatoric uncertainty in simulated and real-world settings.

P2: Evaluation of uncertainty methods does not account for their individual components. To accurately assess the capabilities of an uncertainty method, improvements must be traceable to its individual components C0-C3 (see section 6.2). When a variation of one component is proposed, its interaction with the remaining components must also be examined. Only such rigorous analysis enables the identification of genuine scientific progress and fosters a deeper understanding of uncertainty estimation in segmentation.

Current practice: A common pattern in current literature is to highlight a potential improvement in one component while neglecting the others, often by evaluating in a single simplified setting. For instance, Gonzalez et al., 2021 investigate a specific uncertainty measure (C2) that does not require aggregation, yet apply a simple mean aggregation to all baselines—an approach that can be strongly influenced by the number of foreground pixels. This makes it unclear whether the reported improvement stems from the proposed C2 or from the potentially ill suited aggregation strategy of the baselines (C3). Similarly, Czolbe et al., 2021 use a "sum aggregation" for AL, which might result in querying larger objects.

Another frequent pattern is reporting only pixel-level downstream tasks while overlooking image-level tasks that inherently require aggregation. Examples include studies reporting only calibration (G. Wang et al., 2019; Gustafsson et al., 2020; S. Hu et al., 2019; Postels, Segu, et al., 2021) or pixel-level failure detection (G. Zhang et al., 2022; Mehta et al., 2020). However, one of the main aims in failure detection is to identify and defer faulty subjects or inputs for e.g. human analysis, which questions a plausible application for deferring individual pixels.

In contrast, Jungo et al., 2020 avoid P2 by explicitly studying and ablating individual components of uncertainty methods. Despite this exception, a broader reflection by the community on validation practices is needed to overcome this pitfall at scale.

→**Proposed Solution:** By evaluating the influence of each individual component of an uncertainty method (C0-C3) while holding the remaining components fixed, one can clearly identify which specific modifications lead to measurable performance improvements and whether an improvement with regard to some component is an improvement that holds even under different selection of components.

P3: Evaluation of uncertainty methods restricted to too few downstream tasks. Beyond theoretical studies and claims about separating uncertainty types, it is essential to recognize that uncertainty estimation is not an end in itself. It must serve clearly defined purposes, which should be validated on real-world applications. For practitioners to determine whether a given uncertainty method is suitable for their specific task, it is crucial that proposed methods are evaluated across a broad spectrum of downstream tasks.

Current practice: In the current literature, most studies validate uncertainty methods on only a single downstream task such as OoD-Detection (Lambert et al., 2022; Holder and Shafique, 2021), failure detection (G. Zhang et al., 2022; Mukhoti, van Amersfoort, et al., 2021; Mobiny et al., 2021),

AL (Mackowiak et al., 2018; Colling et al., 2020; S. Xie et al., 2020), calibration (G. Wang et al., 2019; Gustafsson et al., 2020; S. Hu et al., 2019; Postels, Segu, et al., 2021; Mehrtash et al., 2020), or ambiguity modeling (S. Kohl et al., 2018; Monteiro et al., 2020). Also, some task formulations are limited in scope, such as failure detection purely on i.i.d. test data not considering failure sources from potential distribution shifts which represent a major source of failure cases. This shortcoming that has been studied recently for classification tasks (Paul F Jaeger et al., 2023). More generally, the conceptual foundations of a proposed uncertainty method are rarely tied to a single application. Studying their broader usability across multiple downstream tasks is therefore highly relevant to the community. The current practice of sparse task validation thus poses a major challenge for practitioners seeking to identify the most appropriate method for their specific problem.

→**Proposed Solution:** Our evaluation protocol encompasses OoD-Detection, AL, failure detection, calibration, and ambiguity modeling, covering both biomedical 3D datasets and street scenes with 2D natural images. These two settings represent safety-critical domains in which high-stakes decisions rely on accurate segmentations.

6.4 Empirical Study

6.4.1 Design of Uncertainty Separation Study.

In this comprehensive separation study, our primary focus is to investigate the ability of uncertainty measures to effectively distinguish between AU and EU, a claim frequently made in theoretical works (A. Kendall and Gal, 2017b). Recognizing that uncertainty measures are often associated with specific uncertainty types (see eq. (2.10)), we evaluate different prediction models (C1) to determine whether their corresponding uncertainty measures (C2) successfully capture the uncertainty types they are theoretically claimed to represent. We formalize the task of separating aleatoric (AU) and epistemic (EU) uncertainty by posing four guiding questions (Q1-Q4):

In this comprehensive separation study, we investigate whether uncertainty measures can effectively distinguish between aleatoric and epistemic uncertainty, a capability that is commonly asserted in theoretical literature (A. Kendall and Gal, 2017b). Given that uncertainty measures are typically designed to capture specific uncertainty types (see eq. (2.10)), we systematically evaluate different prediction models (C1) to assess whether their associated uncertainty measures (C2) successfully quantify the uncertainty types they theoretically claim to represent. We formalize the task of separating aleatoric (AU) and epistemic uncertainty (EU) through four guiding questions (Q1–Q4):

Q1 Do AU-measures capture AU? **Q2** Do EU-measures capture AU?

To assess the ability of uncertainty methods to capture aleatoric uncertainty, we employ normalized cross-correlation (NCC) as a quantitative metric, comparing predicted uncertainty maps against reference uncertainty maps derived from inter-rater disagreement (for details, see section B.1). We complement this quantitative evaluation with qualitative inspection of the uncertainty maps. According to theoretical expectations, successful separation of uncertainty types implies that aleatoric uncertainty measures should demonstrate both high NCC values and high qualitative fidelity (Q1 = “yes”), while epistemic uncertainty measures should not exhibit these characteristics (Q2 = “no”).

Q3 Do EU-measures capture EU? **Q4** Do AU-measures capture EU?

To evaluate the ability of uncertainty methods to capture epistemic uncertainty, we assess their capacity to distinguish cases with induced distribution shifts, which are associated with epistemic uncertainty, from independent and identically distributed (i.i.d.) cases. We employ the AUROC ranking metric at the image level for this evaluation. Since the spatial manifestation of epistemic uncertainty within images cannot be known a priori, we omit qualitative inspections for this analysis. According to theoretical expectations, successful separation of uncertainty types implies that epistemic uncertainty measures should achieve high AUROC values (Q3 = “yes”), while aleatoric uncertainty measures should yield low AUROC values (Q4 = “no”).

Dataset details. We conduct this separation study across multiple datasets: a synthetic toy dataset, the LIDC-IDRI (LIDC) dataset with two metadata-based distribution shifts, and the

GTA5/Cityscapes (GTA5/CS) dataset. Detailed information on the specific dataset configurations is provided in section 6.4.3.

6.4.2 Design of Downstream Task Study.

Through this study, we aim to comprehensively understand the performance and capabilities of various uncertainty methods in practical settings. The study is conducted on the LIDC dataset, including two metadata shifts, and the GTA5/CS dataset. Specifically, we evaluate the following downstream tasks (see section B.1 for task definitions and metric details):

1) Active Learning (AL). After training on the i.i.d. training set, uncertainty methods are employed to query the top-k most uncertain images from the OoD training set. This setup simulates two scenarios: transfer learning, or a model that performs well on specific sample subgroups while exhibiting weak performance on others. We measure the effectiveness of uncertainty methods in selecting informative samples by computing the relative improvement in Dice score between initial training (starting budget) and subsequent training (incorporating queried samples), normalized against the performance gain achieved through random sampling.

2) OoD-Detection (OoD-D). For OoD-Detection, we assess whether images exhibiting a distribution shift from the i.i.d. training data produce higher uncertainty values. This task is defined at the image level for two reasons: first, in semantic segmentation, the source of distributional shift is not necessarily localized to a specific spatial region (except in cases of novel class appearance); second, human evaluation of distribution shift requires assessment of the entire image. Performance is measured using AUROC.
Connection to AL: An uncertainty method that fails to identify data shifted from the i.i.d. training distribution will be unable to select samples that improve model performance on the shifted data, representing a critical failure mode for active learning.

3) Failure Detection (FD). This evaluation measures how well uncertainty values correlate with segmentation quality. We employ the area under risk coverage curve (AURC) (Geifman et al., 2019; Paul F Jaeger et al., 2023) and the excess area under risk coverage curve (E-AURC) (Geifman et al., 2019), adapted for segmentation tasks by defining risk based on Dice score rather than accuracy. While AURC jointly evaluates ranking quality and overall segmentation performance, E-AURC focuses primarily on ranking quality through comparison with an oracle ranking derived from the segmentation quality itself. We perform this evaluation on both the i.i.d. test set and the OoD test set.
Connection to AL: The ability of an uncertainty method to identify failure cases is crucial for AL, as such cases are most likely to yield substantial updates to model decision boundaries when incorporated into training.

4) Ambiguity Modeling (AM). This task evaluates the ability of uncertainty methods to model and quantify label ambiguity at the pixel level. We measure both the normalized cross-correlation (NCC) (S. Hu et al., 2019) and the Generalized Energy Distance (GED) (S. Kohl et al., 2018) between segmentation outputs and reference segmentations to assess sample diversity.
Connection to AL: If an uncertainty method can properly model the inherent ambiguity of samples, it should enable AL methods to downweight highly ambiguous images during query selection. This is beneficial because samples with high label ambiguity are generally less informative for training than samples where uncertainty stems primarily from model limitations rather than data quality.

5) Calibration (CALIB). We assess the reliability of pixel-level uncertainty estimates by measuring the Average Calibration Error (ACE) (Neumann et al., n.d.; Jungo et al., 2020) in combination with Platt scaling, following (Paul F Jaeger et al., 2023).
Connection to AL: This metric evaluates how well pixel-level uncertainties indicate pixel-level errors. Because Platt scaling can achieve good calibration when the underlying ranking is sound, ACE primarily reflects the quality of uncertainty ranking rather than absolute uncertainty magnitudes.

6.4.3 Experimental Setup

Utilized datasets This section provides an overview of all datasets and highlights key aspects, particularly our approach to addressing pitfall 1 (P1) by inducing both aleatoric and epistemic

uncertainty. For comprehensive dataset details, see section B.2. Both real-world datasets are partitioned into i.i.d. and OoD subsets. Models are initially trained exclusively on the i.i.d. subsets and subsequently evaluated on both i.i.d. and OoD subsets across all downstream tasks.

Toy dataset. We generate a synthetic 3D dataset containing spheres and cubes as segmentation targets. To induce aleatoric uncertainty, we apply Gaussian blur to the object boundaries and simulate three annotators with distinct segmentation styles along these blurred borders. To induce epistemic uncertainty, we introduce distribution shifts in the geometric properties of objects—including color, shape, and position—and incorporate background noise to mitigate shortcut learning (Geirhos et al., 2020). Since this synthetic dataset is specifically designed to address the research questions posed in the separation study (see section 6.4.1), we define the following training and testing scenarios:

Scenario 1. Q1 + Q2: Training models on data with induced aletatoric uncertianty; testing on i.i.d. data also containing aleatoric uncertainty

Scenario 2. Q3 + Q4: Training models on data without ambiguity; testing on i.i.d. data and shifted data

Scenario 3. Q4: Training models on data with aleatoric uncertianty; testing on (**a**) i.i.d. data and shifted data without aleatoric uncertainty and (**b**) i.i.d. data with aleatoric uncertainty (blur) and shifted data without aleatoric uncertainty (blur) (see section B.2.1 for details)

LIDC-IDRI (LIDC). To study uncertainty methods in a real-world setting, we employ the LIDC-IDRI dataset (Armato III et al., 2011), with the task of segmenting lung nodules in $64 \times 64 \times 64$ crops from 3D CT volumes, following S. Kohl et al., 2018. Each nodule has been annotated by four different raters, enabling us to directly induce aleatoric uncertainty through inter-rater disagreement. We include only nodules annotated by all four raters, which serve as the aleatoric uncertainty reference.
We further induce epistemic uncertainty by designing two metadata-based distribution shifts: textured (i.i.d.) versus non-textured (OoD) nodules (texture shift; LIDC TEX) and benign (i.i.d.) versus malignant (OoD) nodules (malignancy shift; LIDC MAL). Since epistemic uncertainty is defined by reducible model uncertainty, we verify that performance on the shifted samples is substantially lower than performance achieved when these samples are included during training, confirming that the model can learn to reduce this uncertainty.

GTA5/Cityscapes (GTA5/CS). We employ the synthetic GTA5 (Richter et al., 2016) and real-world Cityscapes (Cordts et al., 2016a) driving datasets jointly, as both share the same semantic class taxonomy. To induce aleatoric uncertainty, we follow S. Kohl et al., 2018 by randomly swapping label pairs with probability $\frac{1}{3}$ from <class> $\rightarrow$ <class 2>, representing a semnatically identical class, for: “sidewalk”, “person”, “car”, “vegetation” and “road”. Epistemic uncertainty is induced through a synthetic-to-real distribution shift, where models are trained on simulated GTA5 data (i.i.d.) and evaluated on real-world Cityscapes data (OoD).

6.4.4 Studied Uncertainty Methods

We now detail the specific instantiations of each component (C0–C3) used in our experimental evaluation.

C0: Segmentation Backbone. For the synthetic and LIDC datasets, we employ the 3D U-Net architecture (Ronneberger et al., 2015), a well-established model widely used in medical image segmentation. For the GTA5/CS dataset, we use HRNet (Jingdong Wang et al., 2020), which has achieved state-of-the-art performance on Cityscapes. While we maintain a fixed backbone (C0) throughout our main experiments, this choice can be varied in future studies. Implementation details are provided in section B.3.1.

C1: Prediction Model. We evaluate five distinct prediction models:

1. A deterministic softmax model (Softmax).
2. A model trained with Monte Carlo dropout, where dropout remains active during inference to enable posterior sampling (Dropout/TTD)(Gal and Ghahramani, 2016).

3. An ensemble of five independently trained models, interpreted from a Bayesian perspective (Ensemble) (Lakshminarayanan et al., 2017).
4. A softmax model applying test-time augmentation (TTA) (Ayhan and Berens, 2018).
5. A Stochastic Segmentation Network (SSN) (Monteiro et al., 2020), which explicitly models aleatoric uncertainty by learning to generate multiple plausible segmentations through a latent variable representing segmentation variability.

Implementation details are provided in section B.3.2.

C2: Uncertainty Measure. We evaluate several uncertainty measures theoretically associated with predictive uncertainty, epistemic uncertainty, or aleatoric uncertainty. For the deterministic Softmax model, we use only the maximum softmax response (MSR) as a measure of general predictive uncertainty, computed as $1 - \text{MSR}$. For Bayesian models (Dropout and Ensemble), we employ predictive entropy as a PU measure, mutual information $I[Y; \theta \mid x]$ as an epistemic uncertainty measure, and expected entropy as an AU measure (see section 2.3). For the TTA model, which introduces the random augmentation variable T, we interpret predictive entropy as predictive uncertainty, mutual information $I[Y; T \mid x]$ as epistemic uncertainty, and expected entropy as aleatoric uncertinaty (see section B.5). For the SSN, which employs a latent variable Z to model label variability, we interpret predictive entropy as predictive entropy, expected entropy as epistemic uncertainty, and mutual information $I[Y; Z \mid x]$ as aleatoric uncertainty (see section B.4).

C3: Aggregation Strategy. We evaluate three strategies for aggregating pixel-level uncertainties into image-level scores:

1. *Image-level aggregation*, following (Czolbe et al., 2021; Gonzalez et al., 2021; Jungo et al., 2020), which sums uncertainty scores across all pixels. For images containing a single foreground object, this score correlates directly with object size (see section B.6); consequently, we apply this strategy only to the GTA5/CS dataset.
2. *Patch-level aggregation*, which employs a sliding window of size 10^D (where D denotes image dimensionality) to compute the sum of uncertainties within each window, then selects the maximum windowed sum as the image-level score.
3. *Threshold-level aggregation*, which computes the mean uncertainty only over pixels exceeding a threshold λ (see section B.6 for threshold selection). Since optimal threshold values depend on foreground object size, this strategy is not applicable to the GTA5/CS dataset.

6.4.5 Results of the Separation Study

(a)

Question	Qual./Quant.	Toy Dataset	LIDC	GTA5/CS
Q1: Do AU-measures capture AU?	Quantitative	yes	partially	partially
	Qualitative	yes	often (i.i.d.)	partially
Q2: Do EU-measures capture AU?	Quantitative	no	often yes	no
	Qualitative	no	often yes	no
Q3: Do EU-measures capture EU?	Quantitative	yes	yes	yes
Q4: Do AU-measures capture EU?	Quantitative	depends on AU in data	partially	mostly not

(b)

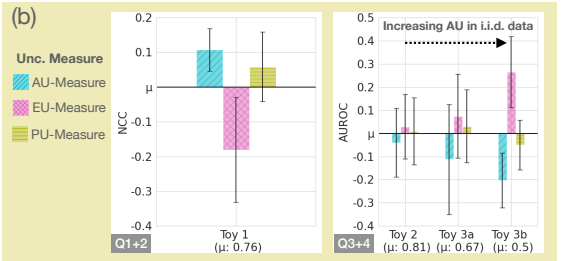


Figure 6.2: **a)** Summary of separation study findings. Green indicates agreement with theoretical expectations, red indicates disagreement, and orange indicates partial agreement. **b)** Quantitative results on the synthetic dataset underlying panel (a). Results are shown for each uncertainty measure (C2), aggregated across prediction models (C1) and aggregation strategies (C3). Corresponding results for LIDC and GTA5/CS datasets are presented in fig. 6.3 (indicated by gray-shaded "Q" markers). Complete details are provided in section B.7. Figure is taken from Kahl, Carsten T. Lüth, et al. (2024b).
Abbreviations: AU, aleatoric uncertainty; EU, epistemic uncertainty; PU, predictive uncertainty.

The main findings of the uncertainty separation study are summarized in fig. 6.2a, while detailed quantitative results are presented in fig. 6.2b for the toy dataset and in fig. 6.3 for LIDC

and GTA5/CS, where gray-shaded "Q" indicators mark the corresponding research questions. A comprehensive analysis, including qualitative evaluation of uncertainty maps, is provided in section B.7.

Modeling aleatoric uncertainty (Q1 and Q2). Aleatoric uncertainty measures capture aleatoric uncertainty more effectively than epistemic uncertainty measures on the toy dataset, though this pattern is less consistent on real-world datasets. On the LIDC dataset, where aleatoric uncertainty arises from inter-rater disagreement primarily at nodule boundaries, the distinction between aleatoric and epistemic uncertainty measures is not evident in NCC scores. For GTA5/CS, where induced label ambiguities affect entire spatial structures, aleatoric uncertainty measures generally outperform epistemic uncertainty measures. Notably, the absolute NCC scores of aleatoric uncertainty measures vary considerably across prediction models on all datasets. We attribute this variation to SSNs capturing the full spatial extent of label ambiguities, while other models tend to overemphasize boundary regions.

Modeling epistemic uncertainty (Q3 and Q4). Epistemic uncertainty measures consistently capture epistemic uncertainty more effectively than aleatoric and predictive uncertainty measures across all datasets. However, the magnitude of this advantage depends strongly on the amount of aleatoric uncertainty present in both training and test data. Specifically, when i.i.d. data contains higher aleatoric uncertainty, the performance gap between epistemic uncertainty measures and other measures widens, as separating ambiguity in the i.i.d. setting enables epistemic uncertainty measures to focus on truly epistemic sources of uncertainty. This effect is particularly evident on GTA5/CS, where spatially extensive label ambiguities result in stronger epistemic uncertainty measure performance compared to LIDC.

General insights. Although both aleatoric and epistemic uncertainty measures generally behave according to theoretical expectations, the degree of achievable separation depends on dataset characteristics, particularly the presence and spatial distribution of ambiguities in i.i.d. and OoD cases.

As theoretically motivated in section B.5, TTA is most effective for modeling epistemic uncertainty, resolving an ongoing debate in the literature. This conclusion is based on our proposed epistemic uncertainty measure for TTA, which exhibits behavior similar to ensembles and test-time dropout (TTD), often matching or exceeding TTD performance. The comparable performance to ensembles establishes TTA as a computationally efficient alternative for epistemic uncertainty estimation.

For SSNs, the proposed epistemic and aleatoric uncertainty measures perform as theoretically intended on the toy dataset and GTA5/CS. However, on LIDC, the boundary-localized ambiguity appears to be captured predominantly by the epistemic uncertainty measures rather than the aleatoric uncertainty measures.

For AL, these findings demonstrate that uncertainty separation is achievable and should prove particularly valuable for datasets with large sample pools exhibiting strong redundancy, where high aleatoric uncertainty is expected. In such scenarios, proper separation can prevent oversampling of regions that the model has already learned to be inherently ambiguous.

6.4.6 Results of the Evaluation on Downstream Tasks

In this section, we address five fundamental questions essential for practitioners when selecting an uncertainty method. For each downstream task, we analyze the optimal component selection based on our dedicated investigation of *(1) uncertainty type, (2) prediction model, and (3) aggregation strategy*. This is followed by an assessment of the *robustness of our findings across (4) datasets and (5) distribution shifts*. Finally, we synthesize insights for AL obtained through joint evaluation of all downstream tasks, organized by components (C1–C3).

To facilitate interpretation and systematically address these questions across downstream tasks, we isolate the performance of each uncertainty type, prediction model, and aggregation method while averaging over the remaining components. We then visualize the performance of each analyzed

Figure 6.3: Aggregated results relative to mean performance (higher values indicate better performance) for each component (C1–C3) across all experimental settings within each dataset. **Note that high standard deviations are expected because averaging is performed across different component configurations rather than random seeds.** Uncertainty measures inappropriate for specific downstream tasks are excluded from the average and marked with crosses colored according to the respective uncertainty measure. Detailed results are provided in section B.8. Figure is taken from Kahl, Carsten T. Lüth, et al. (2024b).
Metrics: OoD detection (OoD Det) uses AUROC, Active Learning (AL) uses % improvement over random sampling, failure detection (FD) uses AURC, calibration (CALIB) uses ACE, ambiguity modeling (AM) uses NCC.

component relative to the mean performance for each dataset and downstream task. Results are presented in fig. 6.3, with complete detailed results provided in section B.8.

Uncertainty measures unsuitable for specific downstream tasks are excluded from the averaging process for prediction models and aggregation methods, marked with crosses colored according to the respective uncertainty measure. We also report standard deviations across the averaged dimensions. *Note that these standard deviations are expected to be relatively high, reflecting the substantial influence of individual components (C1–C3) on final performance.*

Active Learning. Consistent with theoretical expectations, epistemic uncertainty measures generally outperform predictive uncertainty measures, except on the LIDC MAL task. No prediction model consistently performs above average across all datasets. TTD demonstrates strong

performance on the LIDC datasets, while SSNs excel on GTA5/CS. Ensembles consistently achieve above-average or near-average performance, making them a robust default choice compared to Softmax.

The choice of aggregation method exhibits dataset-dependent variability: patch-level aggregation outperforms threshold aggregation on the LIDC datasets, whereas image-level aggregation yields the best results on GTA5/CS. Overall, surpassing the random AL baseline remains challenging, with only marginal improvements observed on LIDC MAL and GTA5/CS. This finding aligns with recent studies (Mittal, Tatarchenko, et al., 2019a; Mittal, J. Niemeijer, et al., 2023; Carsten Tim Lüth et al., 2023).

OoD-Detection. As theoretically expected, epistemic uncertainty measures consistently achieve above-average AUROC values, generally outperforming predictive uncertainty measures. Among prediction models, ensembles are the only models that consistently perform above average across datasets, followed by TTA, which reliably achieves average or above-average performance.

For GTA5/CS, SSNs outperform other models because their ability to explicitly capture spatially extensive label ambiguities improves their capacity to isolate epistemic uncertainty. The optimal aggregation method appears heavily dependent on dataset properties while simultaneously being crucial for performance. For example, on LIDC MAL, differences between aggregation methods exceed standard deviations, making aggregation the most influential component of the uncertainty method in this case, regardless of modifications to other components.

Failure detection. Since failures can arise from both aleatoric and epistemic uncertainty, predictive uncertainty measures are theoretically expected to perform best. However, our results show that while predictive uncertainty is never the worst performer, it is not consistently the best either.

Epistemic uncertainty performs best overall on the LIDC datasets, while aleatoric uncertainty excels on GTA5/CS. This trend may be attributed to the larger spatial regions of induced aleatoric uncertainty in GTA5/CS, which represent a challenging failure source and increase the importance of aleatoric uncertainty modeling. This again demonstrates that dataset properties are crucial for selecting a well-performing uncertainty method.

Regarding prediction models, ensembles almost consistently outperform others, closely followed by TTA. The choice of aggregation method yields mixed results on LIDC TEX, similar to other downstream tasks. Specifically, on i.i.d. data, threshold aggregation is more effective, while patch-level aggregation performs better on OoD data. On other datasets, trends from other downstream tasks are confirmed.

Calibration. Following the same reasoning regarding failure sources from aleatoric or epistemic uncertainty as in failure detection, predictive uncertainty measures are theoretically expected to perform best, which is clearly confirmed across all datasets. This contrasts with the failure detection results, where alignment with theoretical expectations was less consistent despite the close relationship between tasks. Regarding prediction models, TTD performs at or above average across datasets, while SSNs show strong performance, particularly on the LIDC datasets. This trend remains largely consistent between i.i.d. and OoD data.

Ambiguity modeling. Consistent with theoretical expectations, aleatoric uncertainty measures emerge as the most effective uncertainty measure across all datasets. This trend is particularly pronounced for the spatially extensive ambiguities induced in GTA5/CS.

Regarding prediction models, SSNs outperform all other models in both i.i.d. and OoD scenarios across datasets, which is unsurprising since SSNs are the only model specifically designed for ambiguity modeling. One somewhat unexpected finding is the strong NCC performance of the deterministic model on GTA5/CS.

Consistency across datasets and shifts. When assessing the consistency of best-performing uncertainty methods across datasets, we observe that—beyond expected results such as epistemic

uncertainty excelling in OoD-Detection or SSNs excelling in ambiguity modeling—trends often vary between datasets. This is particularly evident when selecting an appropriate aggregation strategy. Given the low computational cost of evaluating this post-hoc component, we recommend benchmarking different aggregation methods. Comparing i.i.d. and OoD data, the observed patterns appear more stable, although performance on OoD data is generally lower, as expected.

C1: Prediction Model. Ensembles rank among the best-performing methods not only for AL but also across all other downstream tasks, while additionally achieving strong absolute segmentation performance on the i.i.d. test set. This makes ensembles a robust default choice for AL when the computational cost of training multiple models is acceptable. Alternatively, TTA offers a viable option that avoids training multiple models or architectural modifications required by Monte Carlo dropout. Notably, the simple Softmax model consistently ranks among the worst-performing methods, despite being commonly used as the standard prediction model in AL experiments—a trend also observed for OoD detection and failure detection on OoD data.

C2: Uncertainty Measure. The ability to model epistemic uncertainty proves beneficial for AL, as epistemic uncertainty measures generally perform best not only for AL but also for related tasks including failure detection on OoD data and OoD detection. This trend extends to failure detection on i.i.d. data for LIDC, which features a relatively small dataset compared to GTA5/CS, suggesting that a significant source of risk may stem from insufficient training data.

C3: Aggregation Strategy. For AL, patch-level aggregation performs best on the LIDC datasets, whereas image-level aggregation outperforms patch-level on GTA5/CS—a pattern consistent with failure detection on OoD data. For failure detection on i.i.d. data and OoD detection, only LIDC TEX exhibits a different trend.

The influence of aggregation strategy on AL is comparatively smaller than for other tasks, as indicated by the standard deviations, though it should not be underestimated. Since aggregation strategy cannot be directly evaluated in the AL context due to the validation paradox, establishing guidelines for selecting C3 requires further dedicated experiments.

6.5 Discussion

General insights and recommendations. Our empirical study generates the following insights regarding uncertainty methods for semantic segmentation based on our systematic solutions for the observed pitfalls (P1–P3):

P1: Missing evaluation of uncertainty methods regarding the separability of aleatoric and epistemic uncertainty. When testing the feasibility of separating aleatoric and epistemic uncertainty (section 6.4.5), we found that while separation works in toy settings, it does not necessarily translate to real-world data. In examining the actual benefits of separation (section 6.4.6), we discovered that these benefits are heavily dependent on both the downstream task and dataset properties. Therefore, neither the feasibility nor the benefit of separation should be assumed when presenting a new uncertainty method; instead, convincing empirical evidence should be required for both. Our study demonstrates that such rigorous testing resolves prior contradictions in the literature—for example, by disproving the assumptions made in Ayhan and Berens (2018) and G. Wang et al. (2019), revealing that TTA is in fact most suited for modeling epistemic uncertainty rather than aleatoric uncertainty.

P2: Evaluation of uncertainty methods does not account for their individual components. Explicit validation of individual components (C0–C3) demonstrates that in practice, it is essential to select optimal components individually based on dataset properties. One prominent insight is the importance of the aggregation strategy (C3), which can be subject to unwanted correlations and is often oversimplified or neglected in previous work. We show that the choice of C3 is further interdependent with the choices of C1 and C2, and only joint consideration of all components enables finding the optimal method configuration for a given task.

P3: Evaluation of uncertainty methods restricted to too few downstream tasks. Our study enables practitioners to make informed choices for all relevant components on their specific tasks. It also identifies potential pitfalls, such as the fact that SSNs, while excelling in ambiguity modeling, underperform in failure detection. Furthermore, the study identifies ensembles as the

generally most robust method across downstream tasks, while TTA often represents an adequate and computationally efficient alternative.

Research Question: How can systematic analysis of uncertainty estimation for semantic segmentation inform the design of Active Learning query methods?

Our systematic analysis directly addresses the research question by providing concrete, evidence-based recommendations for designing AL query methods for semantic segmentation.

Based on the consistently poor performance of Softmax compared to other prediction models, we recommend using ensembles or TTA as the foundation for AL, as they require no changes to the overall model architecture. Ensembles serve as the default choice since they do not require setting additional augmentation parameters.

Uncertainty measures capable of modeling epistemic uncertainty generally improve AL performance, which likely explains the poor performance of Softmax, as it cannot model epistemic uncertainty. Results from the separation study reveal that the ability to separate epistemic uncertainty is particularly important for datasets with high aleatoric uncertainty where the model is already performant. This finding provides actionable guidance: practitioners should prioritize epistemic uncertainty measures for AL query methods, especially when working with datasets exhibiting significant label ambiguity.

The choice of aggregation strategy should be tailored to dataset characteristics: patch-level aggregation is recommended for datasets with localized structures (e.g., medical imaging), while image-level aggregation is more suitable for datasets with spatially distributed content (e.g., autonomous driving). Since aggregation strategy cannot be directly evaluated in the AL context due to the validation paradox, we recommend benchmarking different strategies on related tasks such as failure detection and OoD detection to inform AL query method design.

Impact. Practitioners can use ValUES to make informed design decisions for uncertainty methods tailored to their specific problems, while methodological developments can be rigorously validated using ValUES, fostering a systematic knowledge base in the field. The ValUES framework is open-sourced and can accessed on `https://github.com/IML-DKFZ/values`.

Chapter 7

Principled evaluation of Active Learning for 3D Biomedical Segmentation

> Information is not knowledge. The only source of knowledge is experience. You need experience to gain wisdom.
>
> Albert Einstein

Research Question: How can we evaluate Active Learning in 3D biomedical segmentation to ensure that measurements of annotation effort reductions are both realistic and transferable to new applications?

Drawing on insights from previous chapters regarding AL for classification and uncertainty estimation for semantic segmentation, this chapter presents an evaluation framework for developing and assessing AL methods in 3D biomedical imaging.

This chapter is based on:

- Carsten T. Lüth, Jeremias Traub, Kim-Celine Kahl, Till J. Bungert, Lukas Klein, Lars Krämer, Paul F. Jaeger, Fabian Isensee, and Klaus Maier-Hein (2025). "nnActive: A Framework for Evaluation of Active Learning in 3D Biomedical Segmentation". In: *Transactions on Machine Learning Research*

The first authorship is shared equally between Carsten Lüth and Jeremias Traub. The project spanned multiple years, with Carsten serving as the overall lead. Carsten was primarily responsible for the writing and literature review, while Jeremias revised the drafts. The experimental framework and analysis were predominantly developed by Carsten, with Jeremias contributing in the later stages. Jeremias executed the final experiments and released the framework.

7.1 Problem Statement

Before we are able to issue a general recommendation for an AL method, first we need to show that it reliably brings performance benefits over computationally cheap annotation strategies like Random sampling in multiple 'realistic scenarios' substantial enough to ensure amortization of the additional costs incured by it during application such as the computational cost for multiple trainings or querying samples from the pool (Carsten Tim Lüth et al., 2023; Munjal et al., 2022a; Mittal, Joshua Niemeijer, et al., 2023).

Whereas multiple studies on AL for natural 2D image and video semantic segmentation exist (Mittal,

Joshua Niemeijer, et al., 2023; Mackowiak et al., 2018), for 3D biomedical imaging there remain many open questions with regard to the effectivenss of AL as findings in 2D do not necessarily translate directly to 3D imaging. This is due to differences in the data itself with 3D biomedical data commonly being highly redundant and featuring a background class occupying most of the image which is a stark contrast to the commonly dense multiclass tasks for 2D semantic segmentation tasks (Cordts et al., 2016b). Further, the annotation cost for biomedical images is much higher commonly requiring specialized personnel and also data viewers. Therefore AL with 3D biomedical image data necessitates making use of efficient annotation strategies only annotation parts of an image such as patches or slices.

As of now, the AL community lacks a common benchmark for the 3D biomedical domain as it is highly fragmented in terms of evaluation practices (see table 7.1) which substantially hinders comparability and aggregation of results across different studies. Most importantly, there is no general consensus on whether employing AL methods leads to reliable performance improvements over Random sampling. The results of many studies indicate that AL methods do not always outperform Random sampling (Nath et al., 2021; Gaillochet et al., 2023a; Gaillochet et al., 2023b; Föllmer et al., 2024; Vepa et al., 2024; Burmeister et al., 2022) and it is also commonly emphasized that Random sampling remains a surprisingly strong baseline (Nath et al., 2021; Burmeister et al., 2022). Burmeister et al. (2022) further conclude that AL methods do not reliably outperform *improved Random sampling strategies* where the sampling is adapted to the 3D structure of the data. Critically, this is the only work investigating improved Random baselines. Moreover, most studies neither employ standardized segmentation models proven to achieve state-of-the-art performance nor use 3D models that explicitly leverage partial annotations during training. Both omissions can substantially reduce overall model performance. Concluding, it is apparent that the current evaluation protocol does not allow for making practically relevant and generalizing assessments based on which a practitioner can make an informed decision whether to employ AL or not.

In response, we introduce a novel framework for evaluating the performance of 3D biomedical AL for semantic segmentation. It systematically addresses shortcomings in prior work, formalized as four pitfalls, by adopting best practices for general AL evaluation and extending them to the specific characteristics of 3D biomedical segmentation. These extensions enable practitioners and developers to more reliably assess the potential performance gains when employing AL in scenarios closely resembling production settings. Concretely, our contributions are:

1. We provide nnActive, a highly configurable AL extension for nnU-Net using partial annotations in the form of 3D patches that ensure state-of-the-art segmentation performance and out-of-the-box adaptation to new segmentation tasks.
2. We introduce Foreground Aware Random sampling as a stronger, more realistic baseline, which tackles the class imbalance typically encountered in 3D images.
3. We perform the largest study to date of AL methods with a specific focus on uncertainty based Query Methods (QMs), encompassing over 7500 nnU-Net trainings on 12 dataset-settings from four different datasets with three respective Label Regimes for each dataset, alongside numerous ablations to allow a holistic view of AL performance benefits.
4. We propose Foreground Efficiency (FG-Eff), a novel metric measuring annotation efficiency which takes into account that annotating background has a negligible annotation effort compared to foreground, setting it apart from other metrics using voxels as a proxy for annotation effort.

7.2 Pitfalls and Solutions for a Systematic Validation of Active Learning Methods in 3D Biomedical Semantic Segmentation

Based on the requirements (R1-R4) stated in section 4.4 for developing a generalizable AL method, we identified four corresponding pitfalls (P1-P4) in the evaluation protocols of related work on AL in the 3D biomedical domain, which impede generalizable and reliable performance assessments. Table 7.1 illustrates the prevalence of these pitfalls in the related literature alongside key design parameters. The intention of concretely highlighting the occurence is not to assign blame but to underscore the importance of rigorous evaluation, as inadequate assessment can obscure which methods are truly most effective, particularly for practitioners encountering AL for the first time.

Table 7.1: Overview of the related work in AL for 3D biomedical image segmentation with regard to the described Pitfalls P1-P4 and key parameters. Retraining indicates whether a model is trained for each AL loop from a standard initialization. ✔ indicates addressed, (✔) partially addressed, and ✗ indicates unaddressed pitfalls. N/A is given, as in the experimental setup, this Pitfall can not occur. N.S. indicates an unspecified value in the manuscript and code. A detailed description of our rating is given in section C.1. Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, Isensee, et al. (2025).

	Query Design	P1	P2	P3	P4	#Datasets	Model	Retraining	#Seeds
Nath et al. (2021)	3D Image	✗	✗	N/A	N/A	2	3D U-Net	yes	5
Burmeister et al. (2022)	2D Slice	(✔)	✗	✔	✗	3	2D U-Net	no	3
Gaillochet et al. (2023a)	2D Slice	(✔)	✗	✗	✗	2	2D U-Net	yes	5
Gaillochet et al. (2023b)	2D Slice	✗	(✔)	✗	✗	1	2D U-Net	yes	5
S. Ma et al. (2024)	2D Slice	✗	✗	✗	✗	2	2D U-Net	no	N.S.
Föllmer et al. (2024)	2D Slice	(✔)	✗	✗	✗	3	2D nnU-Net	no	2
Vepa et al. (2024)	2D Slice	✔	(✔)	✗	✗	3	2D U-Net	yes	5
J. Shi et al. (2024)	2D Slice	(✔)	✗	✗	✗	4	2D U-Net	no	5
Ours	3D Patch	✔	✔	✔	✔	4	3D nnU-Net	yes	4

We address these pitfalls by building the nnActive framework and performing a large scale empirical study adhering to the proposed solutions detailed here.

P1: Evaluation is restricted to too few settings. Evaluating AL methods on a wide variety of datasets and multiple different annotation budgets is crucial to ensure their generalizability. Only by doing so is it possible to obtain generalizing performance estimates, as an AL method must generalize to novel scenarios during application. For example, in practice, it may not be clear what *an adequate annotation budget to avoid cold-start is*, indicated by AL methods being outperformed by Random sampling due to insufficient model performance (Gao, Z. Zhang, Yu, Arik, et al., 2020). Only by evaluating AL over multiple different annotation budgets can the cold-start problem be characterized.

State: Currently, the number of 3D biomedical datasets used for evaluation remains very limited, with more than half of related works relying on only one or two datasets (Nath et al., 2021; Gaillochet et al., 2023b; S. Ma et al., 2024). Moreover, evaluations are typically conducted at a single fixed annotation budget (Nath et al., 2021; Burmeister et al., 2022; Gaillochet et al., 2023a; Gaillochet et al., 2023b; S. Ma et al., 2024; Föllmer et al., 2024; Shimizu et al., 2024). Only Vepa et al. (2024) and Gaillochet et al. (2023a) assess multiple annotation budgets for at least one dataset.

→ **Proposed solution:** We adapt the best practices for AL evaluation proposed by Carsten Tim Lüth et al. (2023) in section 5.2 into a framework for method development and benchmarking, covering a broad range of medical imaging tasks, including multi-organ, tumor, fine-grained, pathological, and non-pathological segmentation. For each dataset, we conduct experiments across three annotation budgets—Low-, Medium-, and High-Label Regime—to provide a comprehensive assessment of AL method performance. In addition, we carry out multiple ablation studies on both the query size and the query patch size.

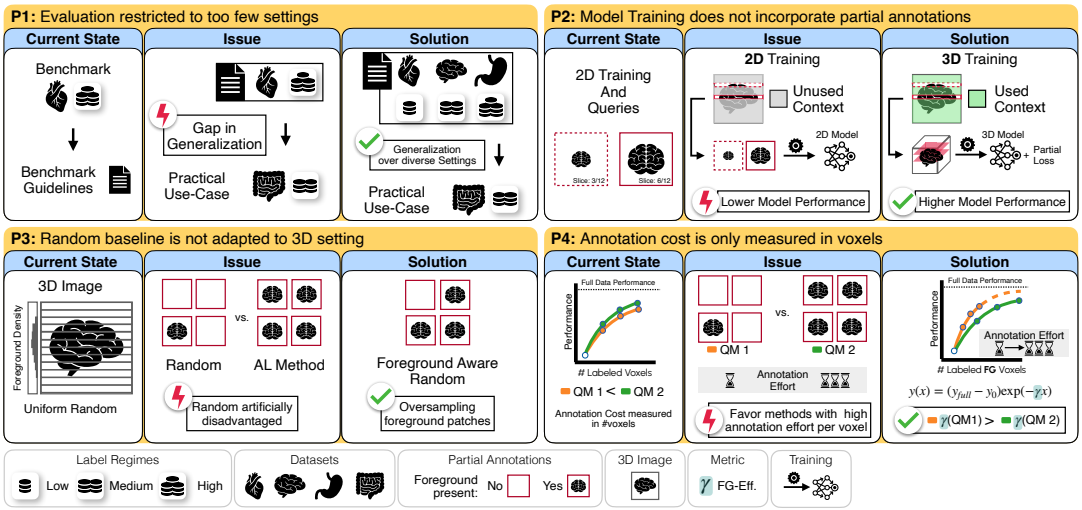


Figure 7.1: Visualization of the four Pitfalls (P1-P4) alongside our solutions and how their presence hinders reliable performance assessments of AL methods for 3D biomedical imaging. For visualization purposes, we use 2D slices as partial annotations.

P2: Model Training does not incorporate partial annotations. For 3D images, image-based query selection is often not feasible given the potentially immense cost of annotating an entire image. Moreover, biomedical datasets typically contain only a small number of individual images. Partial annotations, such as 2D slices or 3D patches (see section C.2 for a definition), are, however, highly informative with respect to the labels of neighboring regions. This stems largely from the strong spatial homogeneity inherent in 3D biomedical images. Leveraging partial annotations for model training while exploiting the unlabeled contextual information substantially reduces the amount of annotated data required to reach performance comparable to training on the fully labeled dataset, without the need for pretrained models or the often compute-intensive semi-supervised training (Gotkowski et al., 2024).

State: Most related works train 2D models on slice-based queries (Burmeister et al., 2022; Gaillochet et al., 2023a; Gaillochet et al., 2023b; S. Ma et al., 2024; Vepa et al., 2024; Shimizu et al., 2024). Training exclusively on these 2D partial annotations discards the surrounding context, reducing overall annotation efficiency. Two strategies which have been used alongside AL to further reduce annotation effort include: Vepa et al. (2024), who train on 2D scribble annotations and use pretrained models, and Gaillochet et al. (2023b) who use Semi-Supervised pretraining. Additionally, several studies conducted on AL for classification and semantic segmentation on nautral images highlight that using well-configured models is another straightforward approach to decrease annotation effort (Carsten Tim Lüth et al., 2023; Munjal et al., 2022a; Mittal, Joshua Niemeijer, et al., 2023; Mittal, Tatarchenko, et al., 2019b). Based on the dominance of nnU-Net in 3D biomedical imaging on both benchmarks and challenges (Isensee, Paul F. Jaeger, et al., 2021b; Isensee, T. Wald, et al., 2024) the work by Föllmer et al. (2024), who proposed an AL integration for nnU-Net that should have state-of-the-art well-configured models. However, their framework only supports 2D training data and queries.

→ **Proposed solution:** We employ 3D nnU-Net models and train them using the partial loss (Gotkowski et al., 2024) in combination with a query protocol that enables 3D patches as queries. A dedicated sampling strategy ensures that these 3D partial annotations are incorporated during training along with sufficient surrounding context. By leveraging the automatic configuration of nnU-Net, we further guarantee that the models are optimally configured for each dataset.

P3: Random Baseline is not adapted to 3D setting. Generally datasets and tasks in 3D biomedical image segmentation feature a background class which occupies most volume of the image and structures of interest (e.g. organ, tumor) that often occupy only a small portion of the images while often also being located in a specific area. Based on this, the standard Random baseline is artificially disadvantaged when combined with partial annotations as it is highly likely to query image regions purely consisting of background or featuring very small regions of the structure of interest, which require minimal annotation effort. For 2D slices this issue was already mentioned by S. Ma et al. (2024). Additionally, specific structures occupy smaller regions in the image than others, leading to a selection bias favoring large structures.

State: Burmeister et al. (2022) evaluate improved Random strategies adapted to the 3D structure of the data, such as stratified sampling (e.g., Strided), and conclude that these improved Random strategies may already be sufficient for many use cases. This finding is particularly concerning, given the surprising strength of the Random baseline (Nath et al., 2021) and that many benchmarked AL methods underperform or only match Random performance (Gaillochet et al., 2023a; S. Ma et al., 2024; Föllmer et al., 2024). This raises the question of how the use of improved Random baselines might have shifted the conclusions of other studies from suggesting that "AL is beneficial" toward a more critical perspective.

→ **Proposed solution:** We employ additional Foreground-Aware Random strategies, which simulate screening an image for a random foreground class and enforce that foreground structures are present in a specified percentage of all queries. Since the primary challenge in manual voxel-wise annotation lies in delineating the structures rather than identifying their approximate location, random image selection with oversampling of foreground regions via random class sampling is a practically feasible approach. This strategy further ensures a diversified class distribution across queries. For details, we refer to section 7.3.

Beyond this, leveraging information about the inherent structure of 3D biomedical tasks may enable the design of multiple improved Random strategies.

P4: Annotation Cost is only Measured in Voxels. Measuring annotation effort of two competing QMs solely on the basis of voxel-based metrics fails to capture substantial differences arising from the structures present within the queries. For instance, a query consisting almost entirely of background with only minimal structures requires little annotation effort but contains the same number of voxels as a query with multiple structures of interest that must be carefully delineated, demanding considerably more effort. Consequently, evaluation methods that rely exclusively on voxel-based metrics for measuring annotation effort can introduce a systematic bias, favoring QMs that result in queries requiring disproportionately high annotation effort per voxel.

State: To our knowledge, none of the related work explicitly accounts for this factor in their measurements or discusses the resulting bias as a potential limitation.

→ **Proposed solution:** We measure annotation efficiency by proxy of the amount of foreground annotation using the decay parameter γ, which we term *Foreground Efficiency (FG-Eff)*. It is derived from an exponential decay fitted to the performance gap between a model trained on the entire dataset and models trained with a limited number of foreground voxels. Higher values of FG-Eff indicate that a QM is more annotation efficient, as it converges more quickly to the performance of a fully trained model (see example in fig. 7.1). By design, FG-Eff only allows for meaningful comparisons of QMs within a single Label Regime and under identical training setups. Since the number of foreground voxels serves only as a proxy for annotation effort, FG-Eff does not replace other performance metrics but should be considered a complementary measure. For further details, including its mathematical definition and interpretation, we refer to section C.4.

7.3 nnActive Framework & Benchmark Setup

The entire benchmark is based on our proposed *nnActive* framework, an extension of nnU-Net for AL with 2D and 3D biomedical semantic segmentation that enables querying 3D patches using AL methods. We focus on 3D patch-based AL to ensure versatility of the framework, as 3D patches can be annotated with multiple strategies, such as dense annotation or sparse slice annotation. The design of nnActive allows seamless integration with the standard nnU-Net framework for both benchmarking[1] and application. This facilitates straightforward implementation of future methodological developments in the benchmarking and application of AL, given that nnU-Net (Isensee, Paul F. Jaeger, et al., 2021b; Isensee, T. Wald, et al., 2024) is widely extended through an ecosystem of projects built directly on top of it (Gotkowski et al., 2024; Roy et al., 2023).

We now introduce the overall design of the *nnActive* framework, together with the benchmark-specific design choices made for our benchmark evaluation.

Model Architecture and Training Strategy. We employ nnU-Net (Isensee, Paul F. Jaeger, et al., 2021b), a self-configuring deep learning framework, as our segmentation model. To extend its standard patch-based training, we enhance the model trainer with region sampling, thereby enriching the observed region with additional unlabeled context from the surrounding image.

Benchmark specific: We use the `3D full resolution` configuration of nnU-Net and train all models for 200 epochs. To increase model robustness, we employ an ensemble of five models trained via 5-fold cross-validation, as ensembles have been shown to improve AL performance by providing more reliable uncertainty estimates (Beluch et al., 2018; Kahl, Carsten T. Lüth, et al., 2024b). For each AL loop, we perform complete retraining of the models, since finetuning can reduce performance (Beck et al., 2021; J. Ash and Adams, 2020), presumably due to the model becoming trapped in a local optimum. While the training of individual models is not seeded, all dataset-related parameters are fixed. Each experiment is averaged over four random seeds.

3D Query Methods. The implementation of QMs for 3D volumetric data in the nnActive Framework consists of two steps. First, for each image all potential queries get a score assigned based on an uncertainty function which is aggregated with a mean aggregation for these patches. Based on these a set of best patches for each image is drawn based on a maximum allowed overlap (o). Second, the final query is drawn from all best patches of the entire training & pool dataset. An example of an uncertainty-based QM is given in algorithm 1.

Benchmark specific: We evaluate the following 8 QMs in our study, described in the following two paragraphs, with no allowed overlap ($o = 0$) between patches.

AL Query Methods. We implemented the following five uncertainty-based AL QMs 1) Predictive Entropy (Settles, 2009), 2) Bayesian Active Learning by Disagreement (BALD) (Houlsby et al., 2011; Gal, Islam, et al., 2017b), 3) PowerBALD (Kirsch, Farquhar, et al., 2023), 4) SoftrankBALD (Kirsch, Farquhar, et al., 2023), and 5) PowerPE (Kirsch, Farquhar, et al., 2023). Both Predictive Entropy and BALD greedily select the top-k uncertainty scores and are therefore referenced as 'Top-k'. PowerBALD, PowerPE, and SoftrankBALD use a top-k selection mechanism with additional noise perturbations, which promotes the diversity of the samples and are therefore referenced as 'Noisy'. More general information regarding query methods is in section 3.1.1.

Benchmark specific: For all QMs, we use mean aggregation where the aggregation size equals the query patch size. The β-parameter for PowerBALD, SoftrankBALD, and PowerPE is set to 1, as proposed by (Kirsch, Farquhar, et al., 2023).

Random Strategies. We use three random strategies as baselines: 1) the standard Random sampling baseline and two more improved Random strategies called **Foreground Aware Random strategies**, 2) Random 33% FG, and 3) Random 66% FG. The Random 66% FG baseline selects 66% of patches oversampling a randomly chosen foreground class by prioritizing regions containing

[1]When extending our results, we emphasize the importance of using our exact nnU-Net version to ensure compatibility.

anatomical structures with foreground oversampling, where half of the patches are centered on a randomly chosen foreground class and the other half are centered on the border of a foreground class. The remaining patches are selected completely randomly. The Random 33% FG baseline operates identically but decreases the proportion of class prioritized sampling to 33%. The addition of these staretgies ensures that Random baselines remain a fair point of comparison as they account for the natural biases present in medical imaging data.

Evaluation Metrics. The general segmentation quality is evaluated using the Mean Dice Score of each 3D image (Dice, 1945). Based on the segmentation quality metric, we use four different metrics, whereby the first three metrics allow relative comparisons of QMs within individual Label Regimes: 1) the Mean Dice score of the final AL loop (Final Dice), 2) the Area Under Budget Curve (AUBC) (Zhan, H. Liu, et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B. Chan, 2022b) aggregating the Mean Dice scores over all AL loops, 3) our proposed FG-Eff measure which is a proxy for the annotation efficiency and 4) the Pairwise Penalty Matrix (PPM) (J. T. Ash et al., 2020), which assesses pairwise performance differences between QMs across multiple annotation budgets, based on a t-test with a significance level of $\alpha = 0.05$.

This combination of metrics enables a holistic assessment of AL methods by capturing absolute performance through Final Dice and AUBC, relative performance via the PPM, and annotation efficiency using FG-Eff. Further details on these metrics and their application in our analysis is provided in section 2.5.3 and section C.4.

7.4 Empirical Study

We give a short description of the experiment setup and a summary of our main findings and their analysis for each of our four ablation studies. Detailed information of the results for each of the four ablations are given in section C.7.

7.4.1 Experimental Setup

We will now provide an overview over the datasets and our specific setup for our experiments in the main benchmark. More details are given in section C.5.

Datasets. Our benchmark spans the following four prominent medical imaging datasets:

ACDC. The automated cardiac diagnosis challenge (Bernard et al., 2018) dataset consists of cardiac MRIs from patients with a range of pathologies, including myocardial infarction, dilated cardiomyopathy, hypertrophic cardiomyopathy, and heart failure with preserved ejection fraction, as well as healthy controls. It provides manual segmentations of the left ventricle, right ventricle, and myocardium at end-diastole and end-systole phases, enabling evaluation of both anatomical and functional cardiac assessment.

AMOS. The multi-modality abdominal multi-organ segmentation challenge 2022 dataset (Yuanfeng Ji et al., 2022) for challenge task 2 consists of CT scans showing the abdominal region. The provided annotations feature 15 organs which enables to assess how segmentation algorithms perform for multiple structures at once.

Hippocampus. The medical segmentation decathlon task 4 dataset focuses on hippocampus segmentation from T1-weighted MRI scans (Antonelli et al., 2022). The annotations delineate the anterior and posterior of the hippocampus, making this dataset particularly valuable for evaluating algorithms aimed at segmenting small, complex neuroanatomical structures.

KiTS. The kidney tumor segmentation challenge 2021 (Heller et al., 2023) dataset comprises contrast-enhanced CT scans of patients with kidney tumors, along with expert annotations of kidneys and tumors. It provides high-quality 3D segmentations for the kidney, kidney tumors and kidney cysts, enabling precise evaluation of automatic segmentation methods.

Preprocessing. Each image is resampled to the median spacing of the dataset, and the data is split into training and pool sets (75%) and a test set (25%), with consistent splits across all seeds and experiments. The nnU-Net framework then generates "fingerprints" from the training and pool sets, which inform the automatic configuration of preprocessing steps, including normalization, patch size, and network architecture. All subsequent preprocessing steps are performed within the nnU-Net pipeline to maintain methodological consistency.

Query Design. The selected query patch sizes for the datasets were selected taking into account the median image size and the size of the structures of interest for the corresponding datasets, leading to the following values: AMOS ($32 \times 74 \times 74$), KiTS ($64 \times 64 \times 64$), ACDC ($4 \times 40 \times 40$), and Hippocampus ($20 \times 20 \times 20$).

Annotation Budgets or Label Regimes. We assess the performance of QMs under three Label Regimes (Low-, Medium-, and High-Label) each corresponding to 5 AL loops to simulate different annotation constraints for each dataset. The entire annotation budget for the Low-, Medium- and High-Label Regimes are defined as follows: 150, 300 and 450 patches for ACDC; 200, 1000, 2500 patches for AMOS; 200, 1000, 2500 patches for KiTS; 100, 200, 300 patches for Hippocampus. We use a starting budget and query size equal to 20% of the full annotation budget of each Label Regime. To ensure that the starting budget is representative for the task, it is allocated to sample random foreground regions of each class, guaranteeing that all classes are present in at least two patches. The remaining part of the starting budget is selected using the Random 33% FG strategy.

7.4.2 Main Study

The evaluation focuses on aggregated results so as to ensure that general on each dataset and across datasets are properly visualized, the detailed results across all datasets are shown in section C.6. The PPM in fig. 7.2 and two Win-/Lose-Barplots in fig. 7.3 show relative performance differences aggregated across all experiments. Meanhile, the ranking of all QMs with regard to AUBC, Final Dice, and FG-Eff for all Label Regimes of each dataset is shown in fig. 7.4 alongside a mean ranking. Based on these aggregated results, we discuss the following five questions with regard to the general performance of AL methods:

How do AL methods compare against Random? We observe that all AL methods consistently outperform Random with respect to performance metrics comparing patch budgets. This is reflected first in the rankings of AUBC and Final Dice in fig. 7.4, where Random consistently ranks among the weakest methods—particularly for Final Dice—and second in the Win/Lose analysis, where all AL strategies outperform Random in over 37% of the evaluated budgets (fig. 7.3a).

At the same time, Random consistently selects the smallest number of foreground voxels, which is reflected in its comparatively strong ranking based on the FG-Eff metric, despite its poor performance in terms of Final Dice and AUBC (fig. 7.4). This raises the question of how much annotation effort is truly reduced when employing AL methods over Random.

How does AL compare against Foreground Aware Random? We observe that Foreground Aware Random strategies frequently outperform AL methods. They also outperform Random across all measured metrics with the exception of FG-Eff where the result is more nuanced. Random 33% FG generally performs slightly worse than most AL methods in terms of both AUBC and Dice (fig. 7.4) as well as when evaluated with the mean PPM (fig. 7.2). Meanwhile, Random 66% FG seems to be the best overall method based on the AUBC mean rank (fig. 7.4) and the positive Win-/Lose-Ratio against all AL methods except for Predictive Entropy (fig. 7.3b). Measured by the mean PPM Random 66% FG is tied with PowerBALD and Predictive Entropy in the second place (fig. 7.2). With regard to the Final Dice, it is, however, apart from Random, the worst performing method as AL methods become better for later AL loops (fig. 7.4).

In conclusion, Foreground Aware Random strategies constitute a substantially stronger baseline than purely Random, and most AL methods have issues outperforming them reliably. This observation

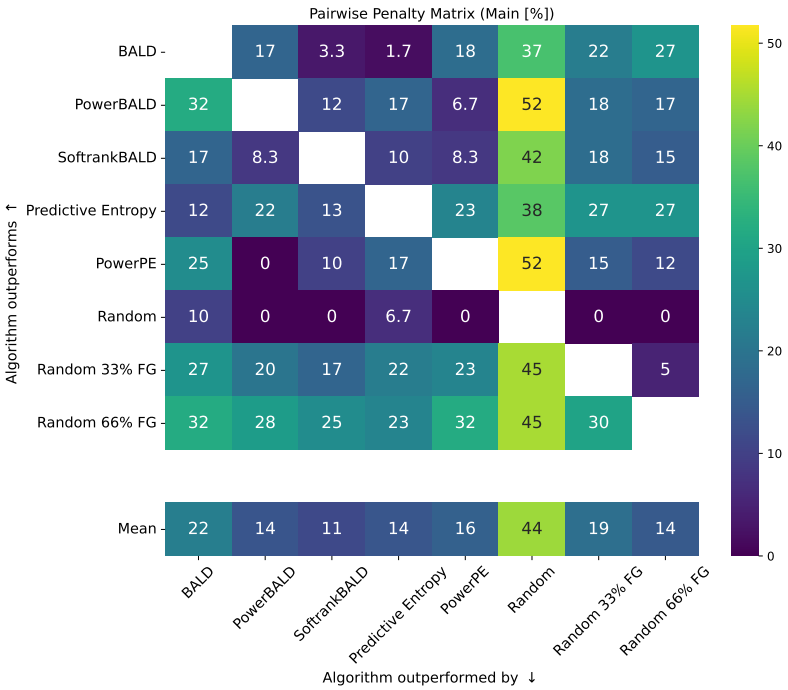


Figure 7.2: PPM aggregated over all experiments of the main study. At each position (i, j) the values indicate the fraction of pairwise comparisons in % where method i significantly outperformed method j. Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, Isensee, et al. (2025).

highlights that the proportion of foreground selected plays a decisive role in the performance of a QM.

Which AL method shows the best performance? Predictive Entropy exhibits strong overall performance across multiple evaluation metrics. It achieves the best mean rank among all AL methods in both AUBC and Final Dice (fig. 7.4), and it is the only method with a positive win/loss ratio against Random 66% FG (fig. 7.3b). In terms of mean PPM performance (fig. 7.2), it ranks jointly in second place together with PowerBALD and Random 66% FG. Performance gains of Predictive Entropy are most pronounced in the later stages of the AL experiments, as indicated by its generally higher ranking with respect to Final Dice compared to AUBC (fig. 7.4).

This behavior, however, also introduces considerable variability in performance, particularly in low-label scenarios where the selected queries are often highly similar. In such cases, Predictive Entropy may even be outperformed by Random sampling (fig. 7.3b). Moreover, its queries frequently target foreground classes, leading to a disproportionately high number of foreground voxels being sampled. This results in relatively low FG-Eff compared to all other methods (fig. 7.4).

How does the dataset influence AL performance gains? We observe substantial differences across datasets regarding the performance gains achieved through AL. When comparing against Random 66% FG on Hippocampus and KiTS, AL is beneficial, whereas the trend is more neutral for ACDC. In contrast, on AMOS all AL methods are generally outperformed by Random strategies, as reflected in the rankings of AUBC and Final Dice in fig. 7.4. In the following, we qualitatively relate these trends to dataset-specific properties.

The primary challenge of **ACDC** arises from the anisotropic spacing and the requirement to precisely delineate three spatially adjacent cardiac structures. These structures occur in both healthy and pathological cases, with most images being cropped to the chest region. The difficulty of achieving exact delineation favors query methods such as top-k QMs or Random 66% FG, which prioritize foreground selection and consequently achieve strong overall performance in terms of AUBC and Final Dice.

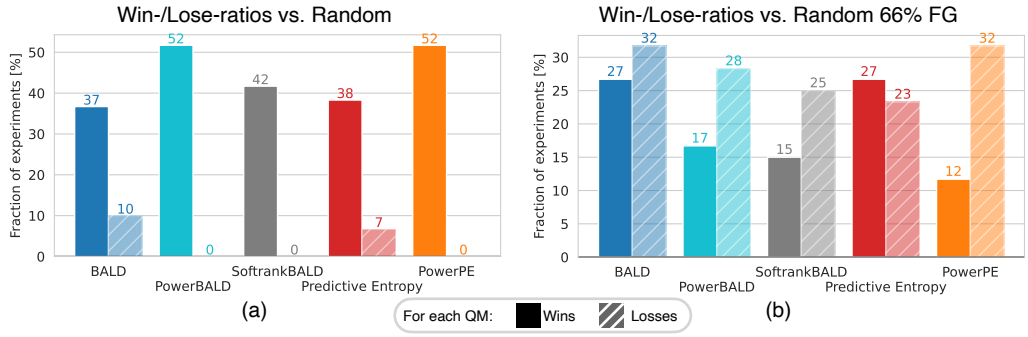


Figure 7.3: A detailed view into the Win-/Lose-ratios of AL methods in the PPM (fig. 7.2) for the main study against Random (a) and Random 66%FG (b). All AL methods outperform Random substantially more often than being outperformed with Noisy QMs, showcasing no Lose-scenarios (a). However, only Predictive Entropy outperforms Random 66% FG slightly more often than it is outperformed (b). Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, Isensee, et al. (2025).

For **AMOS**, the main challenge lies in correctly annotating 15 organs of highly varying sizes distributed across a large anatomical region. We observe that models trained with queries from both Random and AL methods often struggle to reliably segment small organs, such as the adrenal glands, in some cases resulting in a Final Dice score of 0. For Random this limitation arises from the low probability of sampling patches that contain these small structures, whereas for AL methods it is likely caused by redundancy in the queries, which tend to focus on specific classes. This problem is less pronounced for Noisy QMs compared to top-k QMs. In contrast, Random 66% FG and Random 33% FG do not exhibit this behavior leading to them having the overall most stable performance (section C.6.1).

For the **Hippocampus** dataset, the primary challenge is the accurate delineation of the anterior and posterior regions of the hippocampus. Consequently, query methods that greedily focus on borders and regions of high uncertainty, such as BALD and Predictive Entropy, achieve strong performance. As the dataset is cropped to the brain region, the ratio of foreground to background voxels is relatively high, making Random more competitive compared to other datasets; it even outperforms Random 66% FG in the Medium- and High-Label Regimes in terms of AUBC and Final Dice (fig. 7.4). Overall, model performance approaches that obtained when training on the entire dataset (table C.3).

For **KiTS**, the kidney and tumor structures are spatially clustered, while the scans also include large surrounding areas and regions containing only air. Foreground-aware strategies tend to generate many false positives, as their queries predominantly cover foreground regions but fail to capture the full background variability (section C.6.2). This behavior is not observed for AL methods, which also query relevant background areas, resulting in generally higher rankings for AL methods in terms of AUBC, Final Dice, and FG-Eff (fig. 7.4). However, due to the diversity of structures in these scans, top-k QMs can produce redundant queries, causing methods such as Predictive Entropy and BALD to be outperformed in the Low-Label Regime.

What is the influence of the annotation budget on AL Performance? Overall, we observe that low annotation budgets present the most challenging scenario for AL methods, primarily due to potential query redundancy, which is particularly pronounced for the top-k QMs BALD and Predictive Entropy. As a result, these methods rank among the worst-performing approaches in the Low-Label Regime on ACDC, AMOS, and KiTS, especially with respect to AUBC (fig. 7.4). In contrast, Noisy QMs, which select more diversified queries, demonstrate greater robustness and are never outperformed by Random (fig. 7.3a). However, in later AL loops with larger annotation budgets, Noisy QMs tend to underperform relative to their top-k counterparts. For instance, on ACDC, the shift in AUBC rankings between the Low- and High-Label Regimes indicates that redundancy in queries has a reduced impact in later stages (fig. 7.4).

These observations suggest that in early AL loops and under low annotation budgets, Noisy QMs are more reliable than top-k QMs.

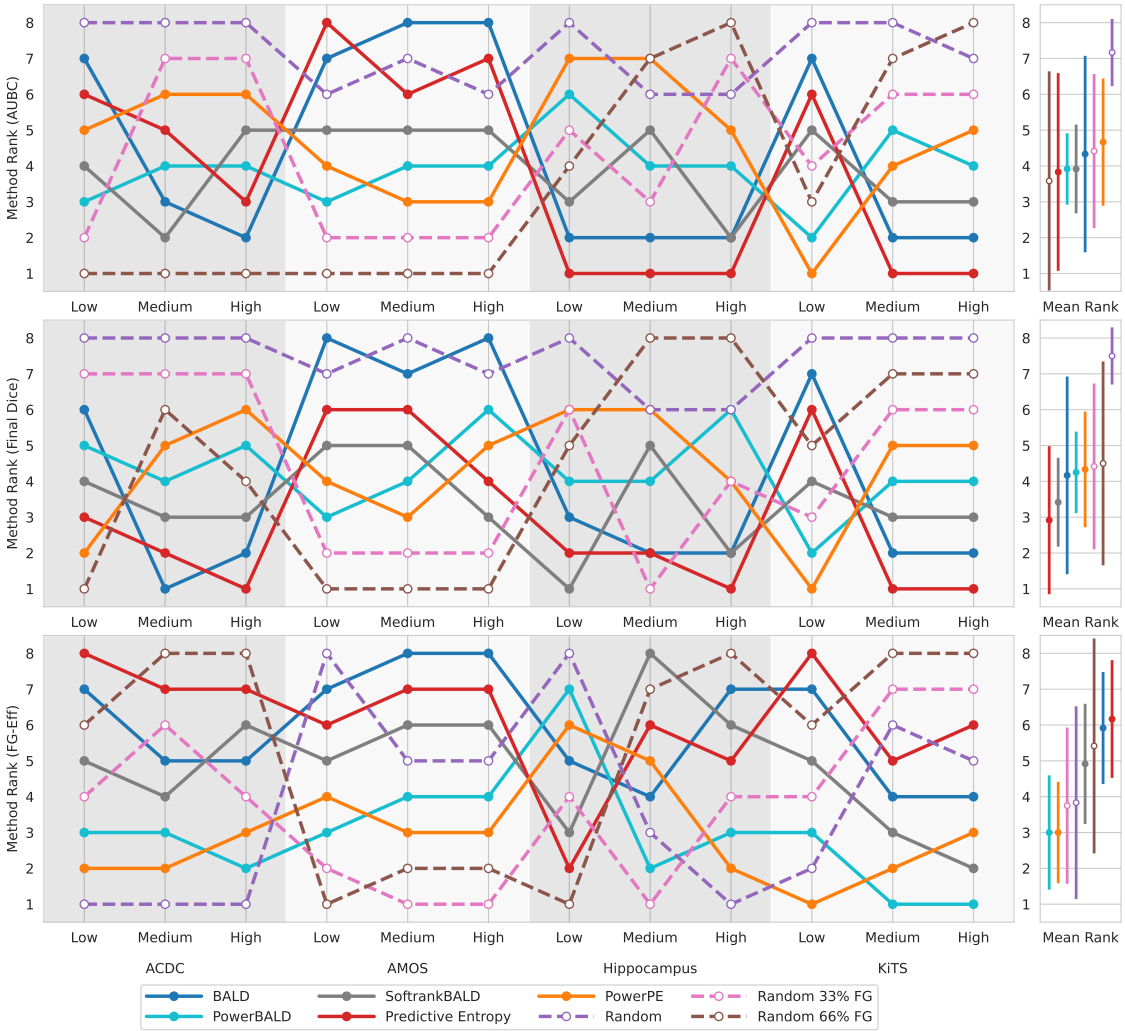


Figure 7.4: Ranking of methods according to AUBC, Final Dice and FG-Eff for each dataset and its Label Regimes (Low, Medium & High) alongside mean with standard deviations (bar).
The trend across datasets with regard to the benefit of AL differs over Foreground Aware Random strategies. On AMOS we observe no benefits when using AL across all Label Regimes whereas on KiTS and Hippocampus AL methods lead to performance improvements and a more neutral result for ACDC. Further, we observe a trend with regard to different Label Regimes where Noisy QMs outperform their top-k counterparts (e.g. PowerBALD and BALD) on the Low-Label Regime. Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, Isensee, et al. (2025).

Table 7.2: **Do smaller query sizes improve the performance of QMs?** Kendall's τ correlations between smaller query size and performance measures. Higher values indicate that smaller query sizes tend to yield better performance. The correlation values range between -1 and 1, where positive values suggest a beneficial effect of smaller queries, while negative values indicate the opposite. A two-sided test was performed with a significance level of $\alpha = 0.1$.
Colorscheme: Significant & positive correleation, positive correlation, negative correlation, significant & negative correlation

(a) Query Size & AUBC

Dataset	ACDC		AMOS		KiTS	
Label Regime	Low	High	Low	High	Low	High
Query Method						
BALD	0.711	0.604	-0.391	0.249	0.249	0.497
PowerBALD	0.178	0.497	0.426	-0.036	0.071	0.533
SoftrankBALD	0.640	0.569	0.462	-0.071	0.178	0.462
Predictive Entropy	0.462	0.711	0.142	0.178	0.391	0.640
PowerPE	0.391	0.355	0.462	-0.142	0.320	-0.142

(b) Query Size & Final Dice

Dataset	ACDC		AMOS		KiTS	
Label Regime	Low	High	Low	High	Low	High
Query Method						
BALD	0.640	0.284	0.107	0.711	0.426	0.178
PowerBALD	0.142	0.213	0.320	-0.107	-0.071	0.604
SoftrankBALD	0.533	0.426	0.391	-0.036	0.142	0.249
Predictive Entropy	0.355	0.569	0.569	0.853	0.391	0.462
PowerPE	0.213	0.426	0.497	0.036	0.320	-0.391

7.4.3 Ablating the Query Size

To assess the impact of query size on AL QMs, we conduct ablation studies using the same absolute annotation budgets as in our main experiments while varying the query sizes. We evaluate three different query sizes of twice the size (QSx2), identical size (QSx1) and half the size (QSx1/2) for one specific starting budget of our main study. This variation results in approximately half or double the number of AL loops, allowing us to analyze how different query sizes influence the performance of AL QMs, separate from other factors. The evaluation is based on two key metrics: the Final Dice score and the AUBC, which is computed only on the overlapping annotation budgets available across all three settings to ensure comparability across different query sizes.

These experiments are conducted on the AMOS, KiTS, and ACDC datasets for both Low- and High-Label Regimes to observe the behavior at the extreme settings.

By analyzing multiple datasets and annotation scales, we aim to gain a comprehensive understanding of how query size affects the performance of AL in different medical imaging contexts through answering the following questions regarding the influence of the query size:

Do AL QMs Benefit from Smaller query sizes? To investigate this, we analyze the correlation between query size and performance using a Kendall's τ (M. G. Kendall, 1948) correlation test on AUBC and Final Dice values. The results, presented in table 7.2, indicate that smaller query sizes consistently improve performance of our benchmarked methods as across all evaluated QMs, we observe significant positive correlations and no significant negative correlations. Generally the effect of smaller query sizes have a strong positive impact on the top-k QMs as they have three significant positive results for both AUBC and Final Dice.

Notably, in the ACDC and KiTS high-budget setting, fewer significant results are observed for Final Dice compared to AUBC, which is counterintuitive given that smaller query sizes are generally expected to provide cumulative benefits. We hypothesize that this occurs because, at high annotation budgets, a substantial portion of the foreground structures in ACDC is already annotated and the performance of the underlying segmentation model is already 'good' – meaning that the decision boundaries does not travel high-density areas of potential queries. As a result, this makes it less likely for larger query sizes to select multiple redundant samples. A similar effect, that for generally larger budgets the benefits of smaller query sizes tend to reduce, has been previously reported by Kirsch, Farquhar, et al. (2023) for object recognition.

How does the Query Size influence rankings of annotation strategies? To investigate this, we analyze the ranking of all annotation strategies for both AUBC and Final Dice with Kendall's τ for each Label Regime and Dataset between pairs of query sizes in table 7.3. Generally, we observe that no ranking is negatively correlated and significant, and that over half of the results are robust (positively correlated and significant).

The rankings for AMOS Low-Label Regime and KiTS High-Label Regime show very little change

Table 7.3: **How robust are method rankings to changes in query size?** Kendall's τ corellations between rankings of QMs with different query sizes. A high value indicates that the rankings between the two settings are similar while lower values denote that they differ. A two-sided test was performed with a significance level of $\alpha = 0.1$.
Colorscheme: Significant & positive correleation, positive correlation, negative correlation, significant & negative correlation

(a) Ranking Correlation AUBC

Dataset	ACDC		AMOS		KiTS	
Label Regime	Low	High	Low	High	Low	High
Setting						
QSx2 vs QSx1	0.571	0.571	0.857	0.857	0.500	0.786
QSx2 vs QSx1/2	-0.214	0.286	0.786	0.929	0.500	0.786
QSx1 vs QSx1/2	0.071	0.714	0.929	0.786	0.857	1.000
Mean	0.143	0.524	0.857	0.857	0.619	0.857

(b) Ranking Correlation Final Dice

Dataset	ACDC		AMOS		KiTS	
Label Regime	Low	High	Low	High	Low	High
Setting						
QSx2 vs QSx1	0.786	0.714	0.786	0.571	0.643	0.929
QSx2 vs QSx1/2	0.286	0.643	0.786	0.500	0.429	0.857
QSx1 vs QSx1/2	0.357	0.786	1.000	0.643	0.643	0.929
Mean	0.476	0.714	0.857	0.571	0.571	0.905

for both AUBC and Final Dice are robust across all compared query sizes. On the AMOS Low-Label Regime, Random FG strategies perform best for all query sizes (see table C.4c), and on the KiTS High-Label Regime, AL QMs like Predictive Entropy perform best (see table C.5b).

Looking at the non-robust settings we observe for the corresponding datasets and Label Regimes, we will elaborate on these changes based on detailed results with rankings shown in section C.7.1.

The strongest ranking perturbation is on ACDC where for the Low-Label Regime with QSx1/2 most AL QMs outperform all Random FG Strategies in terms of AUBC and for the Final Dice leading especially for the AUBC to a strong difference in ranking since the Random FG has the best AUBC for QSx1 and QSx2 (see table C.4a).

Similar behavior can be observed for the ACDC High-Label Regime, where, however, again a change in ranking occurs from smaller to larger query sizes, which favors AL QMs over Random and Random FG strategies in terms of AUBC (table C.4b). For the Final Dice no such trend can be observed and the ranking remains stable.

On the AMOS High-Label Regime, the ranking perturbations stem from increased performance of the Predictive Entropy, especially with regard to the Final Dice leads for smaller query sizes. These lead to its rankings being strongly influenced from the 2nd best ranked strategy for QSx1/2 to QSx2, the 2nd to worst ranked strategy for QSx2 in terms of Final Dice, with similar trends for all other AL QMs, which are more pronounced for the AUBC (table C.4d).

On the KiTS Low-Label Regime the performance changes occur mostly from the QSx1/2 and QSx1 to the QSx2 ranking where for the smaller query sizes PowerPE leads to the best performance in terms of AUBC and Final DICE while for QSx2 it is among the worst performing methods and outperformed by Random 66% FG (table C.5b).

Overall, a change in query size can lead to substantial changes in the ranking of AL QMs relative to random strategies, with top-k QMs especially being affected, swinging from among the best-performing to the worst-performing methods for larger query sizes.

For benchmarking purposes, we believe that reasonably chosen query sizes for a given entire annotation budget, resulting in at least 4 annotation rounds, should suffice, as the correlation between QSx1/2 and QSx1 is significantly positively correlated 5 times out of 6. Especially considering that decreasing the QS by a factor of 2 essentially doubles the compute cost of employing AL the returns are diminishing.

7.4.4 Ablating the Training Length

To assess the impact of the training length on AL QMs we conduct ablation studies using the same setup as in our main study whilst varying the training length. Concretely we evaluate the following three settings of training the model for 500 epochs (500 Epochs), training the model for 200 epochs as in our main study (200 Epochs) and training the models for 500 epochs but using the query trajectories from the models trained with 200 epochs (Precomputed). This design allows us to investigate the effect of longer training while also separating the effects of extended training from its influence on query selections.

Table 7.4: **Does an increased training length of the model lead to better queries?** ΔMetric = Metric(500Epochs) - Metric(Precomputed) for the Training Length Ablation with all models trained for 500 epochs. Larger values show that the queries when training the model for longer are better than from a shorter trained model. Significance comparison performed with a two-sided t-test using a significance level $\alpha = 0.1$.
Colorscheme: Significant & positive difference, positive difference, negative difference, significant & negative difference

(a) ΔAUBC

Dataset	AMOS		KiTS	
Label Regime	Medium	High	Medium	High
Query Method				
BALD	0.98	0.87	1.73	1.64
PowerBALD	0.94	0.52	2.30	1.38
SoftrankBALD	1.00	0.32	1.41	1.04
Predictive Entropy	0.15	0.70	0.58	0.88
PowerPE	0.16	0.46	1.89	1.71

(b) Δ Final Dice

Dataset	AMOS		KiTS	
Label Regime	Medium	High	Medium	High
Query Method				
BALD	1.66	1.12	2.75	2.04
PowerBALD	1.45	0.57	3.92	1.68
SoftrankBALD	1.43	0.45	1.49	0.84
Predictive Entropy	1.76	0.76	2.06	0.44
PowerPE	0.20	0.55	2.18	2.30

The Precomputed experiments are particularly useful in distinguishing whether performance differences arise from the query selection process itself or from the increased training duration.

We performed the experiments on the KiTS and AMOS dataset as ACDC and Hippocampus did not show improvements in Mean Dice when training for more than 200 Epochs on the entire dataset. Our focus is especially on the Medium and High Label Regimes as longer training typically mostly leads to improvements for larger datasets.

By comparing these experimental conditions, we aim to answer the following two questions regarding the relationship between query effectiveness, model training duration, and overall segmentation performance:

Does longer training lead to better queries? We investigate this by comparing the AUBC and Final Dice for all AL QMs of the Precomputed and 500 Epochs settings by computing their differences and testing for statistical significance with a t-test in table 7.4. We observe that the performance metrics for the 500 Epochs models are higher in all cases than for the Precomputed models and with the exception of Predictive Entropy at least in 3 out of 4 settings statistically significant. This indicates that when performance increases with longer training uncertainty based QMs query data more effectively leading to performance improvements even when correcting for performance differences arising from training length.

Ranking based analysis. To evaluate how each of our three training settings influences the ranking of our annotation strategies we perform a Kendall's τ (M. G. Kendall, 1948) correlation test for the AUBC and Final Dice on each Label Regime and dataset between two settings, the results are shown in table 7.5. We deem a ranking as stable when it is positively correlated and significant and will not discuss it except for a change where AL QMs outperform Random strategies where they previously did not or the other way around.

Do gains obtained by using AL persist when training on the queried dataset for longer? Generally, the method rankings between 200 Epochs and Precomputed are stable in 3 out of 4 cases, as shown in table 7.5, with the exception of the AMOS High-Label Regime. We observe on the KiTS dataset that the rankings are stable and the trend that AL outperforms Random and Random FG strategies for both settings. Generally the performance gains of using AL persist from 200 Epochs to Precomputed but decrease in absolute value for the longer trainings on identical queries. This indicates that the results of our ranking for models trained with 200 epochs are likely to hold also for longer trained models on KiTS.

On the AMOS dataset the ranking is stable for the Medium but not for the High Label Regime. Generally observable is a large jump in performance for the models with the AL QMs from 200 Epochs to Precomputed (larger than for Random FG strategies) which we trace back to the Dice

score of specific classes that are hard for the models to learn jumping from 0 to 0.5 for the longer training (see fig. C.7). For 200 Epochs, Random 33% and 66% FG do not exhibit this behavior of individual classes having a Dice score of 0, presumably because they sample more data from these classes (see section C.6.1). On the Medium-Label Regime, PE and BALD have a strong increase in the AUBC, leading them to outperform Random for Precomputed, which they did not do for 200 Epochs, but otherwise, no big changes in ranking. On the High-Label Regime, for the AUBC Predictive Entropy and BALD increase from being outperformed by Random to outperforming Random with longer training and the Predictive Entropy and Softrank BALD outperform Random 33% FG which they did not for shorter training (table C.6b). So, generally, longer training is beneficial for the AL QMs even when the queries are not performed with longer trained models.

Concluding, the gains obtained with AL QMs over Random strategies seem to translate from shorter trained to longer trained models for a shorter time and the performance losses seem to decrease.

How does training length influence the ranking of strategies? For this question the ranking differences between 500 Epochs and 200 Epochs and 500 Epochs and Precomputed from table 7.5 are evaluated.

Generally, the rankings between 500 Epochs and Precomputed showed higher correlation and were more stable than between 500 Epochs and 200 Epochs, being again robust in 3 out of 4 cases for both AUBC and Final Dice, with the exception of AMOS on the High-Label Regime.

For the KiTS dataset there are no changes with respect to the rankings of AL QMs and Random strategies on all Label Regimes and Metrics. The only unstable ranking appears on KiTS Medium for the AUBC comparing the 200 and 500 Epoch Setting, which is mostly due to the inter AL QMs ranking changing with Random and Random FG strategies occupying the worst three ranks in terms of AUBC (table C.6c).

Meanwhile, for the AMOS dataset, the trend is that AL QMs perform better for longer training, which is reasonable as the models guiding the query selection are much better fitted onto the dataset. Most noteworthy for the 500 Epoch setting in the High-Label Regime Predictive and BALD are the only QMs to outperform Random 66% FG in terms of Final Dice which leads to large ranking differences between 500 Epochs and 200 Epochs (which is the only negative correlation) as well as Precomputed (table C.6a). On the Medium-Label Regime, a similar trend can be observed, though not as pronounced, as only Random 33% FG becomes outperformed in terms of Final Dice in the 500 Epochs Setting (table C.6b).

In conclusion, the overall results of the main study with 200 epochs extend to a large degree to 500 epoch settings, indicating that they also should hold for longer training lengths. On the AMOS dataset, this is, however, not the case, as apparently the short training of 200 epochs leads to a systematic disadvantage for the uncertainty-based QMs against the Foreground Aware Random strategies. An optimal AL QM should, however, be able to work under a variety of training settings.

Can the compute cost of AL be reduced using shorter trainings and a final long training? As training is a significant cost factor, this question asks whether we can reduce the training cost while still keeping the gains of AL over Random Strategies? Recalling the Analysis from Q2, it seems that in the scenarios where we obtain large gains from utilizing AL, they should persist while potential performance losses should reduce for the final long training.

However, in Q1 we showed that significant performance differences arise from queries of shorter to longer trained models even when accounting for performance differences due to training length.

In Q3 we observed on AMOS that these differences in query quality can cause the difference between a performance increase over Foreground Aware Random with queries from longer trained models to a performance loss compared to Foreground Aware Random.

For the KiTS dataset, we observed that even though ranking differences among AL methods appeared, the general trend of performance benefits over Random strategies was persistent.

Given this evidence, we suspect that it is likely feasible to perform AL experiments with shorter training. However, one must make sure, by means of validation, that the shorter trained models approximate the task well enough when compared to longer trained models.

Table 7.5: **How does the training length influence method ranking?** Kendall's τ correlation coefficients comparing rankings under different training setups on the AMOS and KiTS for the Medium- and High-Label Regime. Larger values mean rankings are consistent across experiments. Colorscheme: Significant & positive correlation, positive correlation, negative correlation, significant & negative correlation

(a) Ranking Correlation AUBC

Dataset	AMOS		KiTS	
Label Regime	Medium	High	Medium	High
Setting				
Precomputed & 500 Epochs	0.810	0.333	0.714	0.810
200 Epochs & 500 Epochs	0.810	-0.143	0.524	0.905
200 Epochs & Precomputed	0.929	0.500	0.857	0.857

(b) Ranking Correlation Final Dice

Dataset	AMOS		KiTS	
Label Regime	Medium	High	Medium	High
Setting				
Precomputed & 500 Epochs	0.619	0.524	0.810	0.714
200 Epochs & 500 Epochs	0.524	-0.143	1.000	0.810
200 Epochs & Precomputed	1.000	0.500	0.857	0.929

7.4.5 Ablating the Noise strength in Noisy QMs

Our aim is to understand the influence of the noise strength for the Noisy QMs (PowerBALD, SoftrankBALD, PowerPE) in the experimental setup of our main study by an exemplary systematic ablation for PowerBALD.

For PowerBALD β is the parameter which perturbs the ranking of the BALD scores s_{BALD} on a logarithmic scale with Gumbel noise as follows:

$$s_{\text{PowerBALD}} = \log(s_{\text{BALD}}) + \epsilon \tag{7.1}$$

where $\epsilon \sim \text{Gumbel}(0, \beta^{-1})$. The standard deviation of ϵ is proportional to β^{-1}, meaning that smaller values of β introduce greater randomness in query selection, while larger values preserve the original ranking. As $\beta \to \infty$, the ranking remains unchanged after adding noise, whereas as $\beta \to 0$, query selection becomes entirely random. By varying β, we can control the balance between exploration and exploitation in the selection process. It has already been noted by Kirsch, Farquhar, et al., 2023 that in later stages of training the correlation of queries for top-k Methods due to top-k sampling decreases. We suspect therefore that the optimal choice of β will differ across our experiments leaving room for method improvement from the standard setting $\beta = 1$ (Kirsch, Farquhar, et al., 2023) we used in our main study.

To assess the influence of data distribution and Label Regime we perform experiments on the ACDC, AMOS and KiTS dataset for the Low-, Medium- and High- Label Regime whilst varying the parameter $\beta = \{1, 5, 10, 20, 40, \infty\}$ with $\beta = \infty$ being identical to BALD. Generally we only analyze larger values of β as our implementation adds Gumbel noise on the mean aggregated scores leading to the standard deviation of aggregated values naturally being smaller than for singular values.

Using this experimental setting with the results shown in fig. 7.5 we aim to answer the following two questions:

How is optimal β influenced by amount of data? When evaluating the results on each dataset separately, we observe that the optimal parameter of β with regard to the AUBC and Final Dice generally increases from Low to Medium to High Label Regime. This aligns with our broader observations that noise-perturbed QMs generally outperform their top-k counterparts in the early stages of AL but are often overtaken in later loops as training progresses. With regard to foreground efficiency, we observe a steady decrease for higher values of β, converging toward the FG-Eff of BALD across all Label Regimes, indicating that the reduction in queried foreground voxels is greater than the difference in performance.

Generally, the optimal β is therefore correlated with the amount of data and generally increases with more data.

Can the optimal β be selected preemptively? Despite the observation from Q1, we do not identify a single, universally optimal value range of β across all datasets, as they differ greatly across the different datasets. On AMOS, optimal values range from 0 to 5, with a sharp decline in performance for higher values. In ACDC, the optimal range shifts to 5-40, while in KiTS, it spans

1-40. This indicates that dataset properties play an important role in the optimal selection of this parameter, such as – but not limited to – the number of classes and their diversity. Furthermore, we hypothesize the following design decisions of the AL Pipeline to be important: Training length, Query Method (uncertainties and aggregation function) and query patch size.

Based on this, we conclude that setting this value preemptively remains an open question.

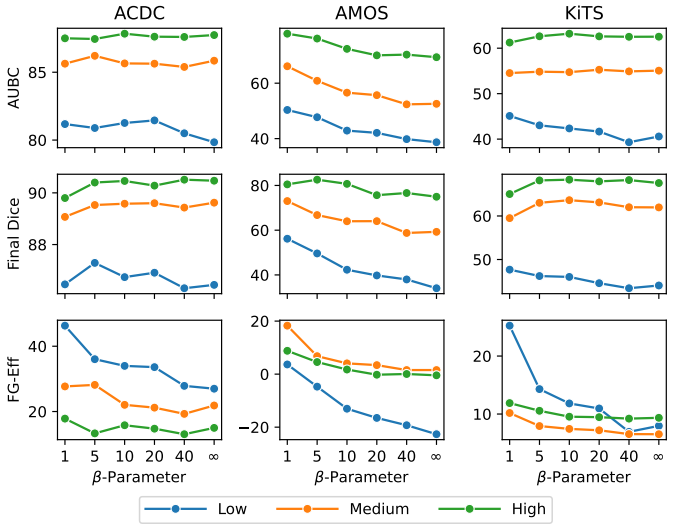


Figure 7.5: **How does the noise strength influence the performance of Noisy QMs?** The β-parameter for PowerBALD plotted against AUBC, Final DICE and FG Eff. for the Low-, Medium- and High-Label Regimes. β-values leading to the best AUBC and Final DICE tends to increase for higher budgets. This indicates that for higher budgets less ranking perturbations perform better. At the same time the FG Eff. decreases which shows that the reduction in perturbation means that more FG is queried.

7.4.6 Ablating the Query Patch Size

Here we aim to understand the influence of the query patch size parameter on our AL experiments.

The query patch size is a hyperparameter of our AL pipeline, setting our work apart as we are the first to allow completely free 3D Patch selection, differentiating our experimental setup from related work, which uses either 2D slice or 3D image queries.

To evaluate its influence, we repeat our entire main study with all four datasets with the respective query patch size halved along each axis whilst keeping the number of patches for each Label Regime identical. We motivate these design decisions as we are interested in seeing whether a more fine-grained selection of areas helps AL methods and the annotation effort for smaller patches does not necessarily decrease linearly with the voxel size.

As the changes with regard to the query patch size make experiments across Label Regimes incomparable, we compare instead across the dataset mean ranking and the overall mean ranking. To do so, we first perform bootstrap sampling to obtain a mean method ranking for each Label Regime of each dataset, which we then aggregate to the dataset and overall level. These mean aggregated rankings are then compared using Kendall's τ (M. G. Kendall, 1948) and a significance test; the results are shown in table 7.6, and the mean ranking values are shown in section C.7.4.

With this setup, we evaluate the following questions:

Does the Query Patch Size influence AL Performance? When comparing the Average Mean rank for Patch$\times$1 and Patch$\times\frac{1}{2}$, it appears that AL has improved Performance compared to Random strategies (section C.7.4), especially with regard to the AUBC. Even though the absolute annotated voxels are reduced by a factor of 16 for Patch$\times\frac{1}{2}$, the trend indicates that the AL methods perform better compared to the Foreground Aware Random strategies on all datasets

Table 7.6: **How does the Query Patch Size influence method benchmarking?** High values indicate that method rankings are consistent across different query patch sizes. Kendall's τ correlation coefficients comparing the mean rankings of each dataset separatively as well the average ranking over all datasets with different patch sizes. A two-sided test was performed with a significance level of $\alpha = 0.1$.
Colorscheme: Significant & positive correlation, positive correlation, negative correlation, significant & negative correlation

	ACDC	AMOS	Hippocampus	KiTS	Average
AUBC	0.357	0.786	0.571	0.793	0.143
Final Dice	0.036	0.764	0.5	0.837	0.714

with the exception of AMOS where for both Patch Sizes the Foreground Aware Random strategies perform best.

When comparing the mean PPMs, we observe that (fig. C.8) similar trends also with the Predictive Entropy being the method with the best win/lose-ratio against Random 66% FG.

In conclusion, we observe that AL methods are surprisingly resilient with regard to the Query Patch Size.

How does the Query Patch Size influence the ranking? The mean rankings of the Final Dice across all datasets are stable, with Predictive Entropy being the best performing method, followed by most other AL methods, with Random FG 66% mixed in between, followed by Random FG 33%, and finally Random as the worst performing. For the AUBC we observe a change in trend for the smaller Query Patch Size where all Noisy QMs are outperformed by their top-k counterparts. We hypothesize that there are two reasons for this behavior: the reduced amount of training data and/or the higher chance of highly similar patterns in the dataset, resulting in high uncertainty values.

On the dataset level, the trend is that for AMOS and KiTS the rankings across Query Patch Sizes are stable, whereas they are less so for Hippocampus and almost completely unstable for ACDC. On ACDC BALD and its derivatives perform better for the smaller than the larger Query Patch Size in terms of AUBC and Final Dice (table C.10). On Hippocampus BALD and SoftrankBALD also perform better for the smaller than the larger Query Patch Size in terms of AUBC and Final Dice, PowerBALD less so presumably due to the noise parameter being too large (table C.12).

We conclude that different Query Patch Sizes can lead to substantial differences in the ranking of QMs.

7.5 Discussion

Research Question: How can we evaluate Active Learning in 3D biomedical segmentation to ensure that measurements of annotation effort reductions are both realistic and transferable to new applications?

To address this research question, we first identified four key pitfalls prevalent in the current evaluation protocols in section 7.2 and systematically proposed solutions that are incorporated into the nnActive framework for semantic segmentation in 3D biomedical imaging, an AL extension of nnU-Net. By increasing the amount of datasets for evaluation, training 3D models on partial annotations, using improved Foreground Aware Random sampling baselines and proposing the FG-Eff metric the benchmark allows for measuring performance estimates of AL methods that are generalizing and practically relevant, which is crucial to draw conclusions regarding performance estimates for real-world application.

By means of our benchmark which is to date the largest empirical AL study in the 3D biomedical segmentation, we obtain the following findings with regard to uncertainty-based AL methods:

- **AL vs. Random:** All evaluated AL methods lead to substantial performance improvements over naive Random sampling, but select substantially more foreground, which likely leads to a higher annotation effort per query.

- **A new Baseline:** Foreground Aware Random sampling is a trivial yet hard to beat baseline. No AL method appears to outperform it reliably.

- **Best AL Method :** Predictive Entropy is overall the best-performing AL method measured by AUBC, Final Dice and PPM, but its performance is highly variable, e.g., for small annotation budgets, and it has the worst overall FG-Eff, which indicates a high annotation effort per query.

- **AL generalization:** AL performance gains strongly depend on dataset and task properties like the ratio of foreground to background and the number of structures to segment.

- **Noisy QMs:** Noisy Query Methods like PowerPE are more reliable in earlier stages of AL and lead to better FG-Eff than top-k Methods like Predictive Entropy.

- **Improving AL performance:** AL method performance can be substantially increased with more compute-intensive settings like longer training and smaller query sizes.

- **AL hyperparameters:** AL method hyperparameters, such as the noise strength, lead to substantial performance differences, but optimal hyperparameters differ between datasets and annotation budgets.

Practical recommendations. Based on the findings listed above, we formulate the following practical recommendations for practitioners and developers:

For practitioners:

1. Based on the strong performance of Foreground Aware Random strategies, we agree with Burmeister et al., 2022 that in many practical scenarios, improved Random strategies, that do not require iterative re-training, may be sufficient.
2. When employing AL, longer training and smaller query sizes represent ways to substantially reduce annotation effort at the cost of more compute.

For developers:

1. Improvements over the naive Random baselines are not sufficient to give a recommendation for widespread use of AL.
2. Method evaluation can be performed using shorter trainings, as performance improvements through longer trainings are consistent across AL methods.

Relevance of our Framework & Benchmark. We believe that the nnActive framework, in combination with our study, will serve as a catalyst for future method development by providing a reliable and unifying benchmark. The entire framework is openly accessible on `https://github.com/MIC-DKFZ/nnActive` and all results of the benchmark are avaliable on `https://huggingface.co/nnActive`. We hope this will lead to wide-spread adoption to the best practices laid out in section 7.2 by overcoming key barriers w.r.t. their adoption which are the high implementation and computational costs required for integrating AL methods into state-of-the-art frameworks due to their complexity and evaluation of multiple AL methods and baselines.

Limitations. Due to the depth and rigor of our evaluation, combined with several orthogonal improvements to the AL experiment design s.a. partial loss and queries in form of freely adaptable 3D patches, we focus our evaluation on uncertainty-based AL methods, which are widely used and generally among the best-performing AL methods for 3D biomedical segmentation (Föllmer et al., 2024) whilst not requiring changes in model architecture and training. We therefore did not evaluate methods like Learning Loss Active Learning (Yoo and Kweon, 2019a), changing the training and diversity-based methods like Core-Set (Sener and Savarese, 2018a). However, our selection of AL methods is still a comprehensive set we believe to be representative of the current state-of-the-art for 3D biomedical AL.

Future directions. Directly building on top of our nnActive framework and study, the following directions are promising: 1) Scaling of diversity-based AL methods like Vepa et al., 2024 and Föllmer et al., 2024 to our performance optimized setting with 3D models and ensembles, as they are, as of now, not represented in our benchmark. 2) Incorporation of Foundation Models for 3D biomedical imaging into our benchmark using nnU-Net due to the decreased time necessary for finetuning and better performance on low annotation budgets. 3) Extension of our proposed FG-Eff metric to a measure which *more accurately* measures annotation effort than number of foreground voxels, e.g. number of clicks for regions (Mackowiak et al., 2018). 4) Incorporation and benchmarking of methods for starting budget selection, as a well-selected starting budget can increase AL performance (Gupte et al., 2024).

Chapter 8

A Simple Uncertainty-Based Active Learning Method for 3D Biomedical Segmentation

All generalizations, with the possible exception of this one, are false.

Kurt Goedel

Research Question: Can we develop an uncertainty-based Active Learning method that reduces annotation effort in Active Learning for 3D biomedical segmentation while generalizing to novel datasets?

As we have shown in the previous section, the efficacy of AL for the 3D biomedical regime with uncertainty based QMs is as of now not necessarily beneficial when compared to improved Random strategies.

Here, our aim is to design a simple and computationally light weight uncertainty based QM which generally performs better or on par with improved Random strategies. For this, we build directly upon the results of the nnActive benchmark in chapter 7.

This chapter is based on:

- Carsten T. Lüth, Jeremias Traub, Kim-Celine Kahl, Till J. Bungert, Lukas Klein, Lars Krämer, Paul F. Jaeger, Klaus Maier-Hein, and Fabian Isensee (2025). "Finally outshining the Random Baseline: A simple and effective solution for Active Learning in 3D biomedical imaging". In: *Submitted to Transactions on Machine Learning Research*. Under review

The first authorship is shared equally between Carsten Lüth and Jeremias Traub. Carsten was primarily responsible for the writing and literature review, while Jeremias revised the drafts. The analysis was predominantly developed by Carsten with scientific input from Jeremias. Jeremias executed the final experiments.

8.1 Problem Statement

This chapter deals with the following problem:

Despite its transformative potential, the effectiveness of AL in reducing annotation costs remains largely unproven for 3D biomedical image segmentation.

This is based on the main result in section 7.5 regarding the efficacy of uncertainty-based AL likely being limited in reducing annotation effort when compared to improved random baseline such as Foreground Aware Random sampling.

Uncertainty-based AL methods represent the most commonly employed group due to their versatility and relative ease of implementation (Munjal et al., 2022a; Gal, Islam, et al., 2017b). Further, uncertainty is a standard building block in more advanced QMs (Hübotter et al., 2024; Föllmer et al., 2024).

Based on this, our aim is to design a simple, yet-effective uncertainty based QM that directly addresses empirically observed shortcomings in the context of 3D biomedical segmentation. To this end, we propose Class-stratified Scheduled Power Predictive Entropy (**ClaSP PE**) which combines two extensions to a standard uncertainty-based AL method:

1. A stratification of standard uncertainty and class-specific uncertainties, which directly addresses the voxel-wise imbalance of classes while still retaining the ability to prioritize hard-to-predict cases.
2. An exponential scheduler for Power-Noising of scores (Kirsch, Farquhar, et al., 2023) which addresses the low diversity of queries especially in early stage AL by perturbing the scores stronger in early AL stages and gradually reducing the noise towards later stages.

In the following we focus on providing compelling that ClaSP PE achieves general annotation cost reductions during application scenarios as it outperforms both standard and improved random baselines in terms of segmentation quality whilst not sacrificing annotation efficiency.

The empirical evidence from our evaluation is delivered in two steps: As a first step, in section 8.3.1, we demonstrate that ClaSP PE consistently outperforms all other AL methods and random sampling strategies on the nnActive benchmark chapter 7, the most comprehensive benchmark to date for AL in 3D biomedical imaging. This encompasses four 3D biomedical datasets, each with three annotation budgets (Label Regimes) that are evaluated with two distinct query designs (query patch sizes), resulting in 24 distinct experimental setups for AL experiments. In the second step, in section 8.3.2, we validate the generalization capabilities of ClaSP PE on four additional datasets by explicitly simulating real-world use-case scenarios (Roll-Out), demonstrating its practical applicability and robustness beyond the benchmark setting. We make sure to set up all parameters for the AL pipeline during Roll-Out according to our *Guidelines for Real-World Deployment* without manual adaptions which can serve as a recipe for practitioners when applying ClaSP PE to novel datasets and tasks.

In summary, our main contributions are:

- We propose ClaSP PE, a simple and effective query method that systematically addresses key limitations of current uncertainty-based AL methods.
- We conduct a large-scale evaluation, demonstrating that ClaSP PE brings reliable performance improvements over standard and improved random sampling baselines for 3D biomedical image segmentation on the nnActive benchmark spanning four datasets and six annotation budgets each.
- We provide evidence for the generalization capability of ClaSP PE by means of a Roll-Out study on four additional datasets to explicitly simulate a real-world use-case with all parameters being set based on our Guidelines for Real-World Deoployment.

8.2 Method

Our proposed query strategy, **Cla**ss-stratified **S**cheduled-**P**ower **P**redictive **E**ntropy **(ClaSP PE)**, is designed to improve AL for 3D biomedical segmentation by effectively balancing informativeness, class representation, and diversity of the queried patches and thereby solves prominent issues of top-k sampling uncertainty methods (as illustrated in fig. 8.1). Starting from a standard **Uncertainty-Based scoring** commonly employed in top-k sampling which returns an uncertainty map $u(x)$ for each image x, we introduce two key modifications: Class Stratified Sampling and an Exponential Scheduler for Score Perturbation. Importantly, these extensions are agnostic to the specific uncertainty scoring function used and can be applied on top of any existing uncertainty-based method.

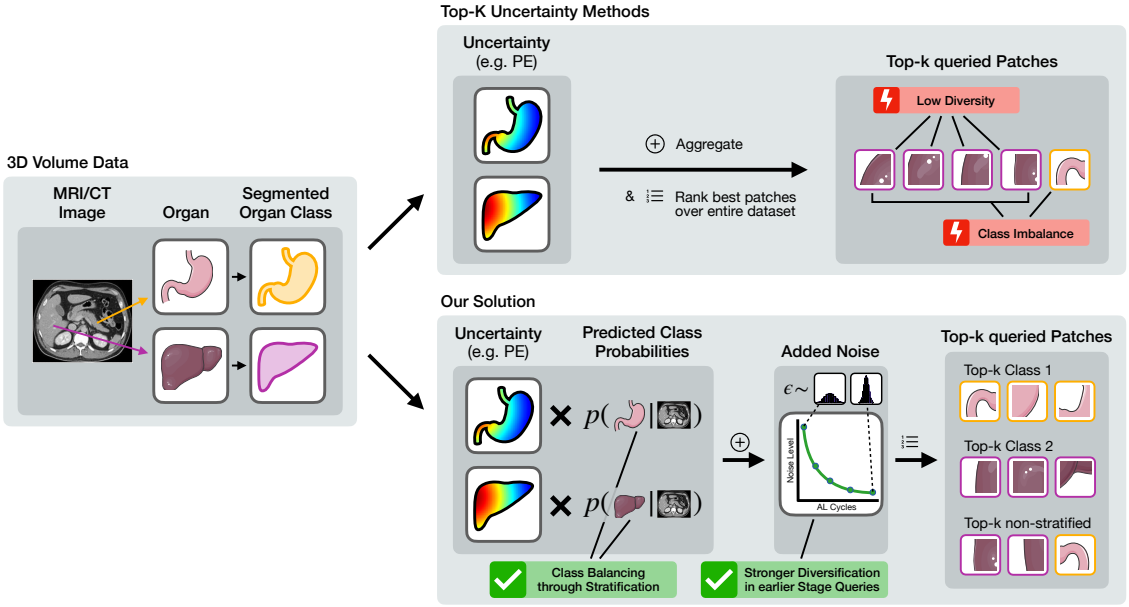


Figure 8.1: **Overview of the ClaSP PE query strategy.** We overcome two key limitations of standard uncertainty-based Active Learning methods (e.g. Predictive Entropy), class imbalance and low diversity of the queries, by adding two simple modifications: (1) class-stratified sampling for 66% of the query budget based on predicted class probabilities, and (2) a scheduler decreasing the noise for score perturbation via log-scale power noising to enhance diversity during query selection. Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, K. Maier-Hein, et al. (2025).

Class Stratified Sampling. To encourage class-balanced selection of queries, we implement a stratified sampling procedure. Specifically, we select an equal number of patches per predicted class based on the model's predictions. For each image x, we compute class-specific uncertainty scores

$$u_c(x) = p_c(x) \cdot u(x), \tag{8.1}$$

where $p_c(x) = p(Y = c|x)$ denotes the predicted probability for class c. Patches are then ranked per class according to $u_c(x)$, and the top N_c patches from each class are selected, where N_c is chosen such that all classes contribute equally to the stratified subset. This ensures that underrepresented classes are not neglected, which naturally supports metrics that average performance across classes (e.g., mean Dice). Importantly, by leveraging the model predictions our approach does not require any additional label information. To our knowledge, balancing queries in this way has not been used in the AL literature before. Crucially, only a fraction α of the samples is selected using this stratified approach, with the remaining $1 - \alpha$ samples being selected based on the standard uncertainty map $u(x)$ to retain sensitivity to highly uncertain examples regardless of class distribution.

An Exponential Scheduler for Score Perturbation via Log-scale Power Noising. To enforce diversity among selected queries, especially in earlier AL cycles, we apply power noising to the scores (on patch-level) before selecting the top-k samples (Kirsch, Farquhar, et al., 2023). Specifically, we perturb the scores on a logarithmic scale by adding Gumbel noise $\epsilon \sim \text{Gumbel}(0, \beta^{-1})$. Additionally, we use an exponential schedule[1] for the perturbation strength β^{-1} such that it decreases towards later AL cycles from an initial value β_0^{-1} to a final value $\beta_{\max}^{-1}$, in order to gradually shift the focus from exploration to exploitation:

$$\beta(t) = \exp\Big(\big[1 - \tfrac{t}{T}\big] \ln(\beta_0) + \tfrac{t}{T} \ln(\beta_{\max})\Big), \quad t = 0, \dots, T \tag{8.2}$$

where t indexes the current AL cycle and T is the total number of AL cycles.

[1]We also experimented with linear and sigmoid schedules but found that exponential schedules generally performs on par or better.

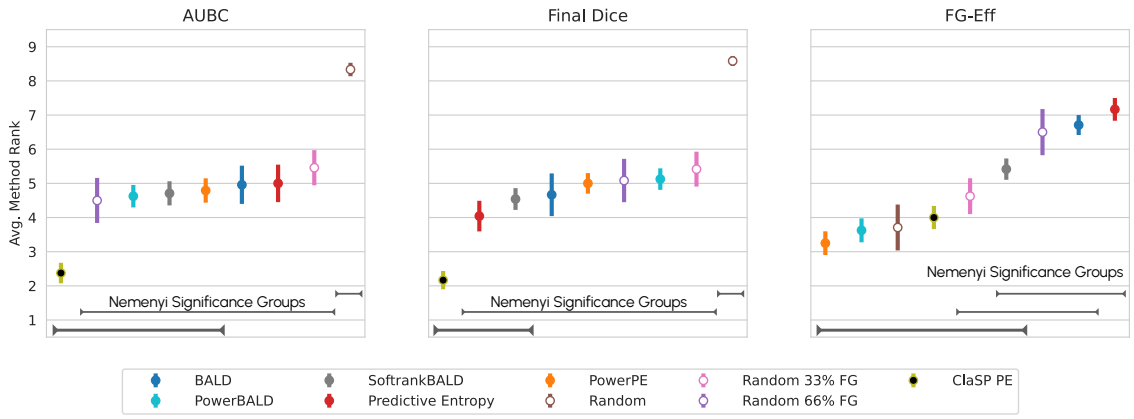


Figure 8.2: **ClaSP PE delivers substantial performance improvements without sacrificing annotation efficiency.** The plots show average method rankings (lower is better) with standard error for AUBC, Final Dice, and FG-Eff across the nnActive benchmark. Results are aggregated over 4 datasets, 3 Label Regimes, and 2 query patch sizes, each evaluated with 4 random seeds, providing robust estimates of method performance. The brackets indicate groups of methods that do not differ significantly based on a post-hoc Nemenyi test at significance level 0.05. Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, K. Maier-Hein, et al. (2025).

For our final method we utilize Predictive Entropy to obtain uncertainty-based scores as we identified it earlier in chapter 7 as the most performant QM. We then apply the stratified selection to $\alpha = 66\%$ of the budget based on our analysis in section 8.3.1. For the exponential scheduler, we use $\beta_0 = 1$ and $\beta_{\max} = 100$ for all experiments.

This method is simple to implement and flexible, yet effective, as our empirical studies in section 8.3.1 and section 8.3.2 demonstrate. We provide an implementation of ClaSP PE in the nnActive framework and a detailed pseudo-code of the method in section D.1.

8.3 Empirical Study

8.3.1 Results on the nnActive Benchmark

We evaluate the effectiveness of our proposed query strategy ClaSP PE on the nnActive benchmark introduced in chapter 7. In total, we conduct more than 1000 nnU-Net training runs across 24 distinct experimental settings (4 datasets $\times$ 3 Label Regimes $\times$ 2 query patch sizes), including dedicated ablation studies. This extensive experimental design covers a broad spectrum of segmentation challenges and provides a solid basis for statistically meaningful conclusions regarding the robustness, efficiency, and generalizability of our method. For clarity, we briefly summarize the crucial components of the benchmark setup to ensure they are easily accessible. Further details can be found in section 7.3 and section 7.4.

Datasets, Label Regimes & Query Patch Sizes. The nnActive benchmark comprises four widely used medical imaging datasets: AMOS2022 (challenge task 2) (Yuanfeng Ji et al., 2022), Medical Segmentation Decathlon Hippocampus (Antonelli et al., 2022), KiTS2021 (Heller et al., 2023), and ACDC (Bernard et al., 2018). Each dataset is evaluated under three Label Regimes (Low-, Medium-, and High-Label), defined by a fixed annotation budget expressed as the number of available patches. In addition, the benchmark includes two query patch sizes: the Main patch size and a reduced variant (Patch$\times\frac{1}{2}$), which is half the size along each spatial dimension.

Baselines. We compare ClaSP PE against the standard Random baseline and two Foreground Aware Random baselines (Random 33% FG and Random 66% FG), as well as five uncertainty-based

query methods: Predictive Entropy (Settles, 2009), Bayesian Active Learning by Disagreement (BALD) (Houlsby et al., 2011; Gal, Islam, et al., 2017b), PowerBALD (Kirsch, Farquhar, et al., 2023), SoftrankBALD (Kirsch, Farquhar, et al., 2023), and PowerPE (Kirsch, Farquhar, et al., 2023). Random 33% and 66% FG simulate the process of selecting a patch around a random foreground region for $X\%$ of their budget. More information regarding can be found in section 3.1.1.

Experimental Setup. Our experimental setup is identical to the nnActive benchmark using four seeds with a fixed test split, and using a custom nnU-Net trainer with 200 Epochs in the 3D full resolution configuration with each AL experiment consisting of 5 cycles. We evaluate AL performance with the following metrics operating on the mean Dice score (Dice, 1945): The Final Dice score achieved after the final AL cycle; the AUBC (Zhan, H. Liu, et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B. Chan, 2022b) which aggregates the mean Dice scores across one AL trajectory over all cycles to measure the overall performance; the FG-Eff (Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, Isensee, et al., 2025), which acts as a proxy for annotation efficiency by setting the performance in relation to the queried foreground voxels by means of an exponential fit; the PPM (J. T. Ash et al., 2020), which quantifies along the entire AL trajectory how often one method significantly outperforms another based on paired t-tests, and can thus simply be aggregated over e.g. datasets. Detailed descriptions of the evaluation metrics is provided in section 2.5.3.

Results. Our evaluation is performed on the highest aggregation level as the goal of AL is to bring generalizing performance improvements for a specific annotation budget. Figure 8.2 shows the method rankings averaged across the nnActive benchmark exact numerical results are provided in section D.4. We find that ClaSP PE achieves the best overall performance in terms of both AUBC and Final Dice, generally outperforming both improved random baselines and established AL methods. Importantly, our approach delivers these segmentation quality gains while maintaining high annotation efficiency, as indicated by FG-Eff: although ClaSP PE does not always achieve top FG-Eff, it consistently ranks among the most efficient methods. This reflects an inherent interplay between segmentation performance and annotation efficiency, where methods that strongly focus on highly informative regions can improve Dice scores but may risk inefficient use of annotated foreground (e.g., Predictive Entropy). Our ablations (see section 8.3.1) further show that score perturbation is crucial for preventing such inefficiencies, and that gradually reducing the noising strength boosts segmentation performance at the cost of only a slight reduction in FG-Eff. Overall, ClaSP PE achieves a favorable balance across this trade-off, providing efficient, informative, and diverse query selection through our proposed modifications.

It is important to note that our baseline models are well adapted to medical datasets by means of proper data augmentation, model architecture and loss formulation, therefore we observe as expected that absolute performance gains for single datasets can be small in absolute value (Mittal, Tatarchenko, et al., 2019a; Carsten Tim Lüth et al., 2023; Beck et al., 2021).

In addition to the average rankings fig. 8.2 shows statistical significance groups obtained using the conservative Nemenyi post-hoc test (Nemenyi, 1963) with a significance level of $p = 0.05$. These groups provide further evidence for the robustness of ClaSP PE as it forms a distinct top-performing group for segmentation performance measured by AUBC and Final Dice and in the large top-performing group for FG-Eff. In contrast, the naive random baseline is consistently ranked lowest for AUBC and Final Dice and is significantly outperformed by all other methods. Among the remaining methods, no statistically significant differences are observed, highlighting that *ClaSP PE is the only method to achieve statistically supported improvements over both random and uncertainty-based baselines*. Overall, ClaSP PE shows the most consistent separation from random and uncertainty-based baselines across all three metrics. Importantly, although SoftrankBALD also appears in the top Nemenyi group, ClaSP PE shows a clearer overall advantage when considering the average rankings (fig. 8.2). Detailed results of the Nemenyi tests are provided in section D.4.

Additionally, when comparing the average Final Dice and AUBC over all settings, ClaSP PE is the only AL method that improves over improved random strategies, as shown in table 8.1. Both PowerBALD and PowerPE outperform their top-k counterparts BALD and Predictive Entropy for the Final Dice performance metric contrary to the rankings in fig. 8.2 which provides further evidence for the more stable performance of these methods across annotation budgets as already

Table 8.1: **ClaSP PE achieves better average performance than both random and AL baselines.** Average Performance aggregated over all 24 distinct AL settings of the nnActive benchmark for AUBC and Final Dice alongside the 95% Confidence Interval (higher is better, indicated by green colorization). Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, K. Maier-Hein, et al. (2025).

Query Method	AUBC	Final Dice
BALD	62.39 ± 0.30	65.43 ± 0.41
PowerBALD	64.81 ± 0.35	67.93 ± 0.29
SoftrankBALD	63.74 ± 0.32	67.32 ± 0.28
Predictive Entropy	63.27 ± 0.40	67.35 ± 0.58
PowerPE	64.85 ± 0.35	68.01 ± 0.38
Random	60.57 ± 0.39	61.65 ± 0.43
Random 33% FG	66.00 ± 0.27	69.74 ± 0.32
Random 66% FG	67.14 ± 0.22	71.14 ± 0.22
ClaSP PE	67.62 ± 0.33	72.81 ± 0.30

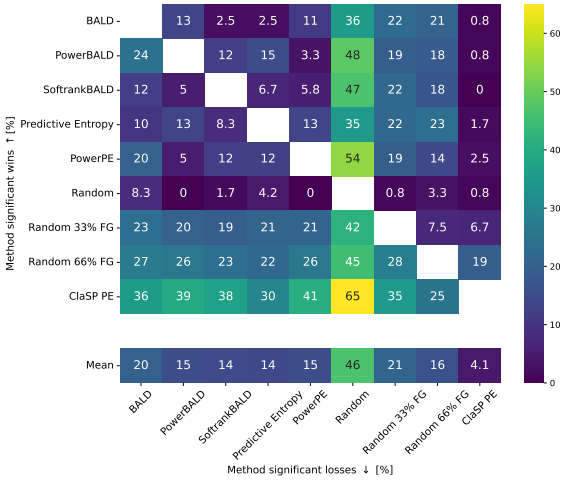


Figure 8.3: **ClaSP PE consistently outperforms both random and active learning baselines across the nnActive benchmark.** The Pairwise Penalty Matrix summarizes statistically significant wins and losses from pairwise t-tests (p=0.05) between methods. Results are aggregated over 24 distinct AL settings on the nnActive benchmark, including 4 datasets × 3 Label Regimes × 2 query patch sizes. Remaining lose scenarios against Random 66% FG stem from challenging Low-Label settings on the AMOS dataset (discussed in section 8.3.1). Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, K. Maier-Hein, et al. (2025).

noted in chapter 7.

ClaSP PE performs well overall and generally delivers substantial performance improvements on the KiTS dataset, as can be seen in table C.3 and table C.8. Hoewever, especially on the AMOS dataset for smaller annotation budgets ClaSP PE underperforms improved random strategies, but shows smaller underperformance compared to the other AL methods. This behavior is further discussed in section 8.3.1.

To complement the aggregate metric rankings and average segmentation performance, fig. 8.3 presents the PPM, assessing pairwise performance differences on the nnActive benchmark. ClaSP PE clearly emerges as the strongest method overall, outperforming all random and AL baselines more frequently than it is outperformed. This underscores the method's robustness and generalizability across diverse settings.

Nonetheless, in roughly 20% of the comparisons, Random 66% FG surpasses ClaSP PE. These cases are concentrated almost exclusively on the AMOS dataset under Low-Label Regimes, a particularly challenging scenario due to the high number of classes and the constrained annotation budget.

Based on this evidence, we note that the combination of score perturbation and stratified sampling substantially boosts the performance of standard Predictive Entropy across all evaluation metrics. Our large-scale evaluation provides clear empirical evidence for the effectiveness and robustness

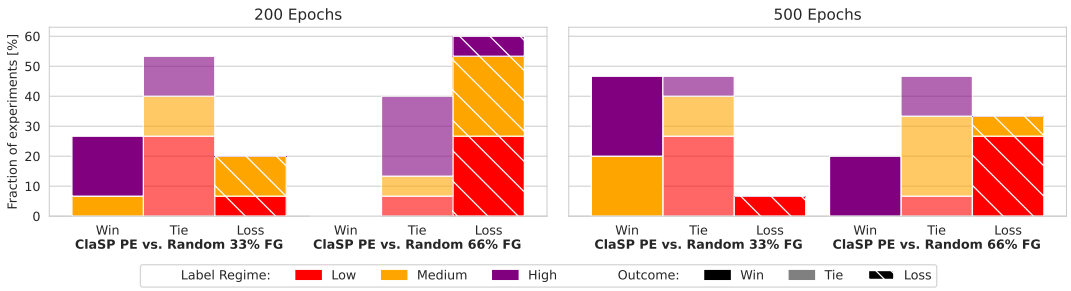


Figure 8.4: **Longer training amplifies the advantage of ClaSP PE over random selection.** Shown are fractions of significant wins, losses, and resulting ties of ClaSP PE against improved random baselines on the AMOS dataset, as computed via the PPM. We compare models trained for 200 (left) and 500 (right) epochs, as well as different Label Regimes (color-coded). While at 200 epochs ClaSP PE loses on 60% of the experiments to Random FG 66% and ties in the rest, whereaes when trained for 500 epochs, it outperforms Random FG 66% in 20%, ties in 48% and loses in only 32%. Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, K. Maier-Hein, et al. (2025).

of these simple yet impactful modifications. Additional qualitative analyses can be found in section D.8.

Investigating Loss Scenarios on AMOS

Based on the insight that the segmentation performance for specific classes on AMOS suffers which can be mitigated by longer training as discussed in chapter 7. We aim to see now whether this pattern is also the main reason for the limited performance gains of ClaSP PE to random baselines on the AMOS dataset. To this end, we conduct an ablation study that evaluates the influence of longer training on AL performance.

Specifically, we compare the performance of ClaSP PE against the improved random baselines (Random 33% FG and Random 66% FG) on the Low-, Medium-, and High-Label Regimes (with a total budget of 200, 1000, and 2500 patches, respectively) using models trained for 200 and 500 epochs. We perform this comparison on the Main nnActive Benchmark, which results in 3 distinct settings.

Generally longer training substantially increases the win-to-lose ratio of ClaSP PE relative to both random baselines. Figure 8.4 shows that in the 500-epoch setting, the number of lose-cases is reduced and primarily confined to the lower Label Regimes. In particular, ClaSP PE now consistently outperforms Random 66% FG in the High-Label Regime, whereas the Low-Label Regime is still dominated by lose-cases. Compared to the Random 33% FG baseline, ClaSP PE shows clear and consistent gains in both the Medium- and High-Label Regimes, underscoring the benefits of extended training. Detailed results are shown in section D.4.4.

These findings suggest that longer training amplifies the advantage of ClaSP PE over random selection but also that foreground focused random strategies still outperform it on the Low-Label Regime on AMOS. We hypothesize that the large number of 15 classes on AMOS makes the Low-Label especially challenging as the annotation budget of 200 patches, when evenly spaced across all classes, captures less than 14 examples per class (compared to 67 on KiTS, for 3 classes). This highlights the sensitivity of AL performance not only to the training dynamics but also to task-specific factors such as the number of classes. Further, we observe in an analysis for AMOS with class-level dice that the loss scenarios on the low-Label Regime mainly stem from the segmentation performance on the right and left adrenal gland which is also less frequently queried compared to Random 66%FG. We show the details in section D.4.5. We therefore emphasize the importance of adapting the annotation budget to the number of classes for practitioners.

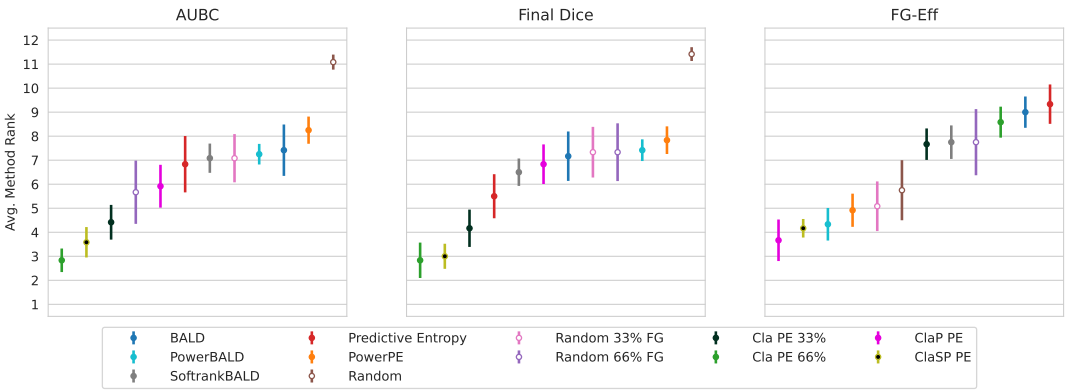


Figure 8.5: **ClaSP PE achieves the best trade-off between segmentation quality and annotation efficiency.** Average method rankings on the nnActive Main benchmark (4 datasets × 3 Label Regimes × 1 query patch size), with additional method variants, Cla PE 66%, Cla PE 33% and ClaP PE. Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, K. Maier-Hein, et al. (2025).

Ablating the Influence of ClaSP PE Components

Our proposed method, ClaSP PE, combines two simple yet effective components: (1) class-balanced sampling applied to a certain fraction of queries, and (2) log-scale power noising applied to the scores prior to top-k patch selection. In this ablation, we analyze the contribution of each component and justify our final design choice. To this end, we evaluate additional method variants, *Cla PE* with $\alpha = 33\%$ and $\alpha = 66\%$ to isolate the effect of class-balanced sampling without power noising and further ablate the fraction of queries for which it is applied as well as *ClaP PE* which is identical to ClaSP PE using $\alpha = 66$ but uses a constant noise value $\beta = 1$ instead of a scheduler. We report their performance across the nnActive Main benchmark.

From the aggregated results, displayed in fig. 8.5, we observe the following: (1) Class-balanced querying improves performance across the board indicated by both Cla PE 66% and Cla PE 33% outperforming standard Predictive Entropy on all evaluation metrics. Moreover, higher stratification rates lead to better segmentation quality as increasing the fraction of stratified queries from 33% to 66% yields improvements in AUBC and Final Dice, with only a minor decrease in FG-Eff. (2) The comparison of Cla PE 66% to ClaP PE shows that the addition of power-noising substantially improves the FG-Eff, indicating improved annotation cost-efficiency through enhanced diversity. However, the power-noising also leads to a reduction in absolute performance measured by AUBC and Final DICE. (3) ClaSP PE leads to the overall best trade-off with regard to annotation efficiency and absolute performance as it is across all three metrics among the best performing which shows the advantage of gradually decayed power noising over no power-noising or power-noising with a fixed rate. This supports the notion that the decaying schedule leads to a more diverse set of queries in early iterations of AL, which gradually become more focused on harder cases when the model has adapted to the data distribution. Detailed results are shown in section D.4.2.

Overall, the combination of 66% stratified querying and gradually decayed power noising provides the best trade-off between segmentation quality and annotation efficiency, justifying the choice of ClaSP PE as our final method.

8.3.2 Simulating Real-World Active Learning in a Roll-Out Study

As already discussed in section 4.4, before a general recommendation with regard to any AL method can be made it is crucial that it is evaluated on a set of held-out datasets in a rollout scenario to ensure that its performance generalizes to novel, previously unseen datasets.

Here, we conduct this exact study across a diverse set of real-world biomedical segmentation datasets. Importantly, we do not perform any dataset-specific finetuning, treating this as a plug-and-play scenario that mirrors how one might apply ClaSP PE in practical, previously unseen tasks.

Baselines The methods we compare include our proposed ClaSP PE, standard Predictive Entropy, which ranked just behind ClaSP PE on the nnActive benchmark, uniform random sampling, and Random 66% FG, a stronger baseline incorporating foreground-aware sampling.

Roll-Out Guidelines We follow all design decisions of the nnActive experiment setup, such as the starting budget and dataset preprocessing, but introduce two new components tailored for real-world deployment: (1) a **systematic selection of query patch size** based on the median connected component sizes of the target structures, and (2) **normalized query budgets**, set to 50 or 100 patches per class depending on task complexity (e.g. the expected homogeneity). These additions ensure that queries remain representative and task-appropriate. Our full Guidelines for Real-World Deployment are provided in appendix D.5.

Datasets. We evaluate performance on four datasets that vary widely in task complexity, number of foreground classes, and annotation difficulty:

LiTS. The liver tumor segmentation challenge dataset (Bilic et al., 2023) consists of 131 contrast-enhanced abdominal CT scans collected from multiple clinical sites with substantial variability in acquisition protocols and patient populations., a two-class foreground segmentation task for liver and tumor.

WORD. The whole abdominal organ dataset (X. Luo et al., 2022) is a large-scale benchmark for abdominal organ segmentation from CT imaging. It is comprised of 120 abdominal CT volumes with annotations for 16 abdominal organs.

Tooth Fairy 2. The tooth fairy 2 dataset is a multi-structure segmentation dataset (Bolelli, Marchesini, et al., 2025; Bolelli, Lumetti, et al., 2024; Lumetti et al., 2024). It consists of 480 CT scans of the head and neck region with annotations for 42 dental structures for different teeth and bones and also including nerves.

MAMA MIA. The MAMA MIA dataset (Garrucho et al., 2025) is multi-center breast cancer benchmark consisting of 1506 sequences of MRI scans with annotations for lesions. The MRI subtraction image is used as image data obtained by subtracting the pre-contrast image from the first available post-contrast image.

Preprocessing. A fixed data split is used for all experiments (75% train & pool, 25% test), which is identical across four random seeds and all images resized to the median size with all following preprocessing steps built on top of nnU-Net. Detailed dataset characteristics are provided in appendix D.2.

Results. The results in Table 8.2 show that ClaSP PE overall performs on par or better than all baseline methods across datasets and metrics. Therefore, it delivers reliable segmentation quality improvements while maintaining or exceeding annotation efficiency, without any task-specific method tuning. While Random shows high FG-Eff on LiTS and WORD, this results from querying only a very small amount of foreground, which artificially inflates FG-Eff without translating into segmentation performance gains as can be seen from the low Final Dice and AUBC values. Predictive Entropy partially shows competitive performance with ClaSP PE in terms of segmentation performance, while ClasP PE demonstrates improved FG-Eff over PE across all roll-out datasets. On the large scale MAMA MIA breast cancer dataset, featuring many redundant structured for a highly complex task, ClaSP PE performs substantially better. Further, the results on the nnActive benchmark (fig. 8.2) reveal that PE fails to reliably outperform random baselines, whereas ClaSP PE shows consistent improvements. Together, these results underscore the robust out-of-the-box performance of the ClaSP PE method and establish it as a practical and effective solution for active learning in real-world 3D biomedical segmentation tasks.

Similarly the PPM shown in fig. 8.6 reveals that ClaSP PE showcases the overall best performance being never significantly outperformed by Random and Random 66% FG while winning in over 50% of all cases and also outperforming Predictive Entropy significantly in 25% of all cases while being significantly outperformed in 5%. We provide detailed results in section D.6.

Table 8.2: **ClaSP PE provides robust performance gains on out-of-the-box deployment.** Performance on the Roll-Out datasets, measured by AUBC, Final Dice, and FG-Eff (higher is better).

Dataset (n_{samples})	LiTS (n=99)			WORD (n=90)			Tooth Fairy 2 (n=360)			MAMA MIA (n=1130)		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Random	51.23	52.38	46.25	77.35	78.03	3.66	61.83	64.32	11.88	55.23	58.24	39.13
Random 66% FG	48.63	50.05	1.27	78.19	78.25	1.34	65.30	68.61	10.85	44.38	45.10	-4.67
Predictive Entropy	57.81	65.38	38.94	78.43	78.96	0.91	66.65	71.97	16.25	59.07	64.74	9.43
ClaSP PE	60.30	65.80	39.60	78.27	78.42	1.33	67.32	71.49	20.07	63.85	68.62	57.36
100% Data Dice		77.3			80.7			72.6			71.0	

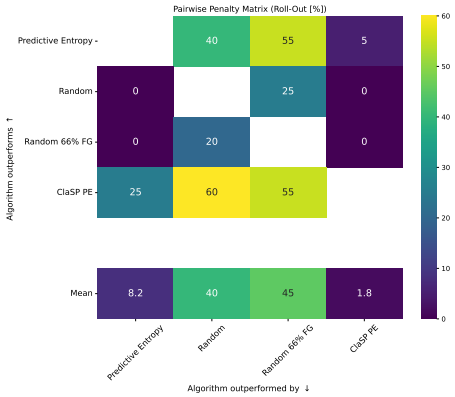


Figure 8.6: **ClaSP PE shows overall strongest performance on the roll-out study.** PPM for the roll-out study aggregated over all settings. In all settings, ClaSP PE wins against or ties with the random baselines. Figure is taken from Carsten T. Lüth, Traub, Kahl, Bungert, Klein, Krämer, Paul F. Jaeger, K. Maier-Hein, et al. (2025).

Limitations

While ClaSP PE demonstrates strong performance across both benchmark and roll-out evaluations, several limitations remain. First, like all AL methods, it faces the risk of benchmark-specific overfitting, due to the necessity of empirically validating design decisions (J. Shi et al., 2024; Föllmer et al., 2024; Gaillochet et al., 2023b; Vepa et al., 2024). Our dual evaluation mitigates this concern but cannot fully eliminate it. Further, as the entire evaluation is based on the average Dice which is the default overlap-based metric for semantic segmentation (L. Maier-Hein et al., 2024), our results do not necessarily extend to boundary-based evaluation metrics or when only specific classes are of interest. Second, the method depends on the predictive capacity of the underlying model: when initial segmentation quality is insufficient, stratified querying becomes less effective, though our guidelines for employing ClaSP PE mitigate this risk, and the use of pre-trained models may further improve early-stage segmentation quality (Gupte et al., 2024). Third, AL is inherently an economic trade-off: reduced annotation cost must be weighed against additional computational overhead, and the optimal balance is context dependent (Settles, 2011). Fourth, while we compared against established strong baselines, more complex AL strategies (s.a. Hübotter et al. (2024) and Föllmer et al. (2024)) could potentially offer further gains, though their adaptability for querying 3D patches remains uncertain. Fifth, ClaSP PE relies on a small set of hyperparameters governing stratification and power-noising. Although validated across diverse datasets, these may benefit from adaptive tuning to better match dataset-specific characteristics. Finally, since our empirical evidence is obtained using the nnActive framework with 3D patches as query design, conclusions may differ under meaningful deviations from it, such as alternative segmentation backbones (Munjal et al., 2022b) or 2D slice queries.

A detailed discussion of these limitations is provided in Appendix D.7.

8.4 Discussion

Research Question: Can we develop an uncertainty-based Active Learning method that reduces annotation effort in Active Learning for 3D biomedical segmentation while generalizing to novel datasets?

We address this research question by proposing ClaSP PE, an AL query method with substantial evidence of reducing annotation effort over random strategies for 3D biomedical segmentation in a close-to-production environment. ClaSP PE adapts a standard uncertainty-based AL query method with class-stratified sampling and exponentially decaying power-noising of scores resulting in consistent performance gains across a wide range of datasets and AL scenarios. As ClaSP PE is conceptually lightweight and easy to implement it can be seamlessly integrated into existing AL frameworks. Its computational cost remains comparable to standard top-k selection methods, making it well-suited for practical deployment.

Relevance for developers and researchers. ClaSP PE can serve as a strong and easy-to-implement baseline for future AL research. Our open-source code and results[2] reduce the experimental overhead for developers and enable fair and reproducible comparisons in methodological studies.

Relevance for practitioners. Our implementation of ClaSP PE offers a solution that can be integrated into real-world annotation workflows. It comes embedded in an AL pipeline that includes guidelines for setting all relevant parameters. This makes it the first plug-and-play AL solution in the 3D biomedical segmentation domain. For real-world deployment, we offer the following recommendations:

- Use ClaSP PE within the nnActive framework, using the auto-configuration of nnU-Net.
- Train models for 1000 epochs, as AL performance generally improves for longer training durations.
- Follow our Guidelines for Real-World Deployment for patch size and query size (see section D.5).

[2]Implementation is at: `https://github.com/MIC-DKFZ/nnActive`
Results are at: `https://huggingface.co/nnActive`

Chapter 9

Discussion & Conclusion

> The greatest and most important problems of life are all in a certain sense insoluble... They can never be solved, but only outgrown...
>
> Carl Gustav Jung

This thesis addresses the theory-practice gap in AL by developing rigorous evaluation methodologies and demonstrating practical annotation savings for biomedical image segmentation. Through systematic investigation spanning image classification and 3D medical segmentation, we establish when AL is likely to reducte annotion cost alongside evidence-based guidance for practical use-case secenarios in 3D biomedical segmentation.

9.1 Summary of Contributions and Research Questions

This thesis makes five primary contributions to active learning research and practice, each directly addressing one of the research questions posed in chapter 1. We present these contributions alongside their corresponding research questions to highlight how our work systematically advances the field.

Contribution 1: Evaluation Requirements for Active Learning

Research Question 1: What are general aspects necessary for the evaluation of Active Learning methods?

In chapter 4, we establish a comprehensive framework for AL evaluation that acknowledges the economic realities of adoption. Our analysis identifies five critical considerations: First, AL evaluation must recognize the economic framework where adoption depends on balancing overhead costs against uncertain annotation savings. Second, evaluators must navigate the *Validation Paradox*—directly validating AL in a use-case scenario contradicts its purpose of reducing annotation effort. Third, effective evaluation requires a decision framework that provides practitioners with expected-value assessments based on their problem characteristics. Fourth, convincing evaluation must demonstrate generalization across settings, show improvements alongside orthogonal baseline optimizations, enable meaningful comparisons, and use effort-aware metrics. Finally, strategic specialization within a clearly defined scope offers a realistic path to credible evaluation without requiring impossible universal generalization claims.

Contribution 2: Evaluation Framework and Pitfall Analysis for Classification

Research Question 2: How can we ensure generalizable and meaningful evaluation of Active Learning methods to guide practitioners in deep active classification?

In chapter 5, we identified and formalized critical flaws in AL evaluation methodology that have plagued the field, presenting them as systematic pitfalls with corresponding solutions. Our framework establishes evaluation principles for deep learning-based AL in classification tasks, emphasizing the importance of comparing AL in combination with orthogonal efficiency techniques (hyperparameter optimization, self-supervised learning, semi-supervised learning) to obtain realistic performance estimates over random sampling. Through systematic application of these principles—using strong, properly tuned baselines; evaluating across multiple datasets and annotation budgets; comparing benefits over random sampling with orthogonal efficiency techniques; and accounting for computational overhead in total cost analysis—we demonstrate that proper evaluation reveals AL's incremental nature. This work shows that AL provides value when baselines are strong but is insufficient to overcome poor model selection, and crucially demonstrates that proper model configuration and training strategies often yield greater benefits than AL query selection alone, providing essential context for understanding AL's role within broader annotation efficiency strategies.

Contribution 3: Uncertainty Estimation Analysis for Segmentation

Research Question 3: How can systematic analysis of uncertainty estimation for semantic segmentation inform the design of Active Learning query methods?

In chapter 6, we conduct a systematic analysis of uncertainty estimation methods for semantic segmentation, addressing three key pitfalls in uncertainty estimation literature that directly inform AL query design. First, regarding separability of aleatoric and epistemic uncertainty, our study reveals that while theoretically feasible in toy settings, separation does not reliably translate to real-world data, and its benefits are highly task- and dataset-dependent. Through rigorous testing, we resolved prior contradictions, most notably clarifying that test-time augmentation models epistemic uncertainty rather than aleatoric uncertainty as previously claimed. Second, component-wise evaluation (prediction models, uncertainty measures, pixel selection, aggregation strategies) demonstrates that optimal configurations must be selected individually based on dataset properties, with aggregation strategy emerging as critically important yet often oversimplified. Third, multi-task evaluation identifies ensembles as most robust across tasks, with test-time augmentations as a lightweight alternative, while revealing method-specific weaknesses (e.g., stochastic segmentation networks excel at misclassification detection but fail at failure detection). For AL specifically, we identify ensembles or test-time augmentations as preferable over commonly employed Softmax models, therefore prioritizing methods that model epistemic uncertainty, especially for datasets with high aleatoric uncertainty where models are already performant.

Contribution 4: nnActive Framework and Benchmark

Research Question 4: How can we evaluate Active Learning in 3D biomedical segmentation to ensure that measurements of annotation effort reductions are both realistic and transferable to new applications?

In chapter 7, building on insights from previous chapters, we develop the nnActive framework for AL deployment in 3D biomedical segmentation. The framework addresses this research question through five key components: (1) domain-appropriate baselines using nnU-Net with standard protocols; (2) partial annotation support enabling realistic cost modeling; (3) improved random baselines accounting for medical imaging characteristics; (4) evaluation across diverse anatomical structures and modalities; (5) effort-aware metrics relating performance directly to annotation volume. This framework integrates best practices from prior chapters while introducing domain-specific adaptations. Through nnActive, we documented previously unreported phenomena in biomedical AL including the superior performance of noise-based query methods in early AL rounds, sensitivity to query method hyperparameters, and strong foreground bias in top-k sampling

strategies—particularly relevant given the extreme class imbalance in medical imaging. Most importantly, we find that there is insufficient evidence to support the use of current uncertainty-based AL methods over improved random strategies.

Contribution 5: ClaSP PE Method and Roll-Out Validation

Research Question 5: Can we develop an uncertainty-based Active Learning method that reduces annotation effort in Active Learning for 3D biomedical segmentation while generalizing to novel datasets?

In chapter 8, we introduce ClaSP PE, an uncertainty-based query method combining class-stratified selection with scheduled power-noise injection, specifically designed to address common failure modes of AL in biomedical segmentation identified on the nnActive Benchmark. Critically, we validate ClaSP PE through roll-out evaluation on four held-out datasets (liver, hippocampus, prostate, lung) spanning different anatomical structures and imaging modalities (CT, MRI), explicitly simulating real-world deployment where no experimental feedback informs method development. ClaSP PE demonstrates consistent annotation reductions across these diverse previously unseen datasets, providing strong empirical evidence that properly designed uncertainty-based AL generalizes to novel biomedical segmentation problems sharing similar characteristics. While we cannot guarantee success on every possible dataset—consistent with the validation paradox—our roll-out evaluation provides the strongest empirical evidence to date for practical AL effectiveness in 3D biomedical segmentation.

9.2 Key Findings

Our systematic investigation yielded several fundamental insights about AL effectiveness and evaluation:

AL Requires Strong Baselines AL does not compensate for poor model configuration or training procedures. Our experiments consistently demonstrate that hyperparameter optimization, pre-trained models, and proper training strategies yield substantially greater performance improvements than AL query selection when starting from weak baselines. AL provides incremental efficiency gains atop already well-optimized pipelines, not a substitute for fundamental ML best practices. This finding has critical implications for evaluation: comparing AL only against random sampling with poorly configured baselines overstates AL's practical value.

Self-Supervised Pre-Training Synergizes with AL Self-supervised learning provides particular benefits in AL contexts. Pre-trained models offer superior initialization, reducing cold-start challenges in early AL rounds when labeled data is extremely scarce. Additionally, faster convergence during AL training loops amortizes the upfront pre-training cost across multiple rounds. Our results demonstrate that SSL and AL are genuinely orthogonal and complementary: SSL provides better starting performance and training efficiency, while AL determines which samples to annotate.

Uncertainty Method Choice Matters for Segmentation The effectiveness of uncertainty-based query methods depends critically on proper uncertainty quantification. Our analysis reveals that ensemble methods provide more reliable epistemic uncertainty estimates than Monte Carlo Dropout for segmentation tasks, directly impacting AL performance. Furthermore, the commonly held belief that test-time augmentation captures aleatoric uncertainty is incorrect—test-time augmentation primarily models epistemic uncertainty, making it suitable for epistemic uncertainty-based query strategies.

Query Method Design Must Account for Domain Characteristics Generic query methods from natural image classification do not transfer directly to biomedical segmentation without domain-specific adaptation. The extreme class imbalance (pathological regions often comprise $< 1\%$ of voxels), foreground bias of top-k sampling, and spatial correlation in 3D volumes all necessitate

specialized approaches. ClaSP PE's class-stratified selection addresses imbalance directly, while power-noise scheduling mitigates sample correlation issues, yielding more robust performance across diverse medical imaging scenarios.

Evaluation Methodology Determines Research Conclusions The pitfalls we identified are not mere technical details but fundamental flaws that can reverse research conclusions. Evaluating AL without proper baselines, without accounting for orthogonal efficiency techniques, or without appropriate metrics leads to overestimation of AL benefits. Our framework establishes that convincing AL evaluation requires: comparison against strong, properly configured baselines; demonstration of benefits when combined with transfer learning and semi-supervised learning; evaluation across multiple datasets and annotation budgets; and use of effort-aware metrics that directly relate performance to annotation cost.

9.3 Implications

For AL Research This thesis demonstrates that AL research must shift focus from proposing novel query strategies in isolation to systematic evaluation accounting for real-world deployment constraints. The field has accumulated hundreds of query methods, yet adoption remains limited because evaluation practices fail to convince practitioners of practical value. Our pitfall analysis and evaluation framework establish minimum standards for credible AL research: proper baselines, orthogonal technique comparison, multi-dataset validation, and effort-aware metrics are not optional luxuries but essential requirements.

Furthermore, our work highlights that AL effectiveness depends critically on the broader ML pipeline context. Future AL research should explicitly characterize the regime where proposed methods provide value, acknowledge computational and implementation costs, and demonstrate benefits alongside—not instead of—established efficiency techniques. The field must move beyond demonstrating superiority over simple random sampling to proving improvements over well-established annotation straties using well-optimized systems.

For Medical Imaging Practitioners Our findings provide actionable guidance for practitioners considering AL adoption. First, ensure baseline optimization: proper architecture selection, transfer learning, hyperparameter tuning, and training procedures should precede AL consideration. These steps already often yield substantial performance improvements, potentially obviating AL need entirely. Second, for 3D biomedical segmentation with high annotation costs, ClaSP PE within the nnActive framework represents a practical, empirically validated approach with open-source implementation. Third, consider alternative or complementary strategies: partial annotations, self-supervised training, and annotation tools such as MedSAM (J. Ma, Y. He, et al., 2024), nnInteractive (Isensee, Rokuss, et al., 2025) may provide comparable efficiency gains with simpler implementation.

Practitioners should view AL as a final optimization step in already well-tuned pipelines, valuable when annotation budgets are severely constrained and baseline optimizations are exhausted. The roll-out validation demonstrates that AL can deliver practical benefits, but success requires domain-appropriate method selection (ClaSP PE for biomedical segmentation), proper implementation (nnActive framework), and realistic expectations (incremental improvements, not transformative savings). We focused on selecting informative patches rather than explicitly evaluating these annotation processes; examining how different techniques interact with patch-based querying remains future work.

On the Importance of Query Design and Annotation Technique. The design of the query, whether it is a whole 3D image, a 3D volumetric patch, a 2D slice, or even a single voxel, substantially impacts the annotation process and tooling efficiency. However, no consensus exists on which query design and annotation process, such as sparse annotation, super-pixels/voxels, or scribbles, is the most economical, as each one has its own advantages and drawbacks depending on the specific task and currently available tooling (Tajbakhsh et al., 2020; Y. Shi et al., 2024). We consider annotation technique selection critical for maximizing economic effectiveness.

Our evaluation uses 3D patches, which support various annotation processes including sparse 2D slice-wise schemes (Çiçek et al., 2016; Burmeister et al., 2022) and scribble annotations (Zihan Li et al., 2024; Gotkowski et al., 2024).

For Evaluation Methodology Beyond AL specifically, this thesis contributes to broader discussions of simulation-based evaluation in machine learning. The validation paradox necessitates reliance on generalization from simulation studies. Our work demonstrates that credible simulation-based evaluation requires: explicit characterization of evaluation scope and limitations; held-out test sets for assessing generalization; realistic modeling of deployment constraints; and transparent discussion of assumptions. Overall these principles extend beyond AL, however they are espcially important due the validation paradox.

9.4 Limitations and Scope

This thesis focuses specifically on pool-based AL for classification and semantic segmentation. We do not address stream-based AL, membership query synthesis, or other AL paradigms. Our segmentation work concentrates on 3D biomedical imaging where 3D patches are used as query design; while we expect insights to transfer to 2D medical imaging and potentially to natural image segmentation, explicit validation on these domains remains future work.

Our evaluation necessarily employs simulation where ground truth labels are known in advance. While we design experiments for maximum realism (including explicit roll-out validation), we cannot fully replicate all real-world deployment aspects: annotator fatigue, label noise, temporal drift, and evolving task requirements. The validation paradox means these limitations are inherent to AL research, not specific shortcomings of our work, but they nonetheless bound our claims' strength.

We evaluate ClaSP PE on medical imaging datasets, representative of typical biomedical segmentation benchmark sizes. However, the roll-out study on held-out test sets could be expanded to more datasets, especially containing tubular structures. Similarly our findings only show that AL is likely to reduce the required annotation but not necessarily that is more cost effective than not employing AL. The question of whether to employ AL or not still remains a cost-benefit analysis based on the annotation cost per sample and additional cost to employ AL.

9.5 Future Directions

Foundation Models and AL The rapid advancement of foundation models for medical imaging (e.g., MedSAM, nnInteractive) creates new opportunities and challenges for AL. These models provide increasingly strong zero-shot and few-shot performance, potentially shifting AL's role from improving weak models to fine-tuning strong models with minimal labeled data. Future work should investigate: (1) how AL query strategies should adapt to foundation model context, where model uncertainty has different characteristics; (2) whether parameter-efficient fine-tuning (LoRA (E. J. Hu et al., 2022), adapters (R. Wang et al., 2021)) interacts differently with AL than full fine-tuning; (3) how to leverage foundation model embeddings for improved query selection. Our framework provides the evaluation methodology for addressing these questions rigorously.

Cross-Domain Transfer and Meta-Learning Our roll-out validation demonstrates within-domain generalization (across different anatomical structures and imaging modalities within medical imaging), but cross-domain transfer (e.g., medical to natural images, or across dramatically different medical imaging tasks) remains challenging. Future work should explore: (1) meta-learning approaches that learn to select query strategies based on dataset characteristics; (2) transfer learning for query methods, where method components trained on source domains adapt to target domains; (3) automated analysis of dataset properties to predict which query methods will succeed. Such work could help practitioners navigate the query method landscape without extensive experimentation.

9.6 Closing Remarks

This thesis demonstrates that AL, when properly evaluated and carefully deployed, can deliver practical annotation savings for biomedical image segmentation. However, AL is not a universal solution to annotation scarcity, nor a substitute for fundamental ML best practices. Rather, it represents a final optimization step in well-engineered pipelines, providing incremental but measurable efficiency gains when annotation budgets are severely constrained and baseline methods are already optimized.

The path to practical AL adoption requires not just better query methods but better evaluation practices, realistic expectations, and domain-specific adaptation. Our contributions—evaluation framework, uncertainty analysis, nnActive toolkit, and ClaSP PE method—provide the methodological foundation and practical tools enabling evidence-based AL deployment in biomedical imaging (see section 9.1). The roll-out validation demonstrates that these advances translate to novel datasets, offering practitioners the confidence that AL is likely to reduce annotation effort in their specific use cases when deployed appropriately. In all this, it is important to acknowledge that all insights regarding the benefits of AL in this work are derived using simulated experimental setups and the extension of these results requires assuming different degrees of generalization. Whilst we strived to make these simulations as realistic as possible, there are natural limitations and we connato guarentee that using our proposed methodology will always yield the lowest overall project cost.

As machine learning continues to advance through foundation models and other innovations, the question "which data should we label?" remains fundamental. Strategic data selection through AL will continue to be relevant precisely because labeled examples defining task solutions are irreplaceable, regardless of architectural sophistication. This thesis provides the evaluation principles, methodological insights, and practical tools necessary to answer that question rigorously how to translate AL from theoretical promise to practical reality for biomedical image segmentation.

Bibliography

Abraham, Alexandre and Léo Dreyfus-Schmidt (Feb. 2021). *Rebuilding Trust in Active Learning with Actionable Metrics*. arXiv: 2012.11365 [cs].

Agarwal, Sharat, Himanshu Arora, Saket Anand, and Chetan Arora (Aug. 2020). "Contextual Diversity for Active Learning". In: *arXiv:2008.05723 [cs]*. arXiv: 2008.05723 [cs].

Ahn, Jiwoon, Sunghyun Cho, and Suha Kwak (2019). "Weakly supervised learning of instance segmentation with inter-pixel relations". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2209–2218.

Ahn, Jiwoon and Suha Kwak (2018). "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4981–4990.

Aklilu, Josiah and Serena Yeung (2022). "ALGES: active learning with gradient embeddings for semantic segmentation of laparoscopic surgical images". In: *Machine Learning for Healthcare Conference*. PMLR, pp. 892–911.

Almeida, Silvia D, Carsten T Lüth, Tobias Norajitra, Tassilo Wald, Marco Nolden, Paul F Jäger, Claus P Heussel, Jürgen Biederer, Oliver Weinheimer, and Klaus H Maier-Hein (2023). "cOOpD: Reformulating COPD Classification on Chest CT Scans as Anomaly Detection Using Contrastive Representations". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 33–43.

Almeida, Silvia D, Tobias Norajitra, Carsten T Lüth, Tassilo Wald, Vivienn Weru, Marco Nolden, Paul F Jäger, Oyunbileg von Stackelberg, Claus Peter Heußel, Oliver Weinheimer, et al. (2024). "Prediction of disease severity in COPD: a deep learning approach for anomaly-based quantitative assessment of chest CT". In: *European radiology* 34.7, pp. 4379–4392.

Amara, Kenza, Lukas Klein, Carsten Lüth, Paul Jäger, Hendrik Strobelt, and Mennatallah El-Assady (2024). "Why context matters in VQA and Reasoning: Semantic interventions for VLM input modalities". In: *arXiv preprint arXiv:2410.01690*.

Angluin, Dana (1988). "Queries and concept learning". In: *Machine learning* 2.4, pp. 319–342.

Antonelli, Michela, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M Summers, et al. (2022). "The Medical Segmentation Decathlon". In: *Nature communications* 13.1, p. 4128.

Armato III, Samuel G., Geoffrey McLennan, Luc Bidaut, Michael F. McNitt-Gray, Charles R. Meyer, Anthony P. Reeves, Binsheng Zhao, Denise R. Aberle, Claudia I. Henschke, Eric A. Hoffman, Ella A. Kazerooni, Heber MacMahon, Edwin J. R. van Beek, David Yankelevitz, Alberto M. Biancardi, Peyton H. Bland, Matthew S. Brown, Roger M. Engelmann, Gary E. Laderach, Daniel Max, Richard C. Pais, David P.-Y. Qing, Rachael Y. Roberts, Amanda R. Smith, Adam Starkey, Poonam Batra, Philip Caligiuri, Ali Farooqi, Gregory W. Gladish, C. Matilda Jude, Reginald F. Munden, Iva Petkovska, Leslie E. Quint, Lawrence H. Schwartz, Baskaran Sundaram, Lori E. Dodd, Charles Fenimore, David Gur, Nicholas Petrick, John Freymann, Justin Kirby, Brian Hughes, Alessi Vande Casteele, Sangeeta Gupte, Maha Sallam, Michael D. Heath, Michael H. Kuhn, Ekta Dharaiya, Richard Burns, David S. Fryd, Marcos Salganicoff, Vikram Anand, Uri Shreter, Stephen Vastagh, Barbara Y. Croft, and Laurence P. Clarke (2011). "The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A Completed Reference Database of Lung Nodules on CT Scans". In: *Medical Physics* 38.2, pp. 915–931.

Ash, Jordan and Ryan P Adams (2020). "On Warm-Starting Neural Network Training". In: *Advances in neural information processing systems* 33, pp. 3884–3894.

Ash, Jordan T., Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal (Feb. 2020). "Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds". In: *arXiv:1906.03671 [cs, stat]*. arXiv: 1906.03671 [cs, stat].

Atighehchian, Parmida, Frédéric Branchaud-Charron, and Alexandre Lacoste (June 2020). "Bayesian Active Learning for Production, a Systematic Study and a Reusable Library". In: *arXiv:2006.09916 [cs, stat]*. arXiv: 2006.09916 [cs, stat].

Ayhan, Murat Seckin and Philipp Berens (2018). "Test-Time Data Augmentation for Estimation of Heteroscedastic Aleatoric Uncertainty in Deep Neural Networks". In: *Medical Imaging with Deep Learning*.

Bachman, Philip, Ouais Alsharif, and Doina Precup (2014). "Learning with pseudo-ensembles". In: *Advances in Neural Information Processing Systems* 27.

Baumgartner, Christian F, Kerem C Tezcan, Krishna Chaitanya, Andreas M Hötker, Urs J Muehlematter, Khoschy Schawkat, Anton S Becker, Olivio Donati, and Ender Konukoglu (2019). "Phiseg: Capturing Uncertainty in Medical Image Segmentation". In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II 22*. Springer, pp. 119–127.

Baumgartner, Michael, Paul F Jäger, Fabian Isensee, and Klaus H Maier-Hein (2021). "nnDetection: a self-configuring method for medical object detection". In: *International conference on medical image computing and computer-assisted intervention*. Springer, pp. 530–539.

Bearman, Amy, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei (2016). "What's the point: Semantic segmentation with point supervision". In: *European Conference on Computer Vision*. Springer, pp. 549–565.

Beck, Nathan, Durga Sivasubramanian, Apurva Dani, Ganesh Ramakrishnan, and Rishabh Iyer (2021). "Effective Evaluation of Deep Active Learning on Image Classification Tasks". In: *arXiv preprint arXiv:2106.15324*. arXiv: 2106.15324.

Beluch, William H., Tim Genewein, Andreas Nurnberger, and Jan M. Kohler (June 2018). "The Power of Ensembles for Active Learning in Image Classification". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT: IEEE, pp. 9368–9377.

Bengar, Javad Zolfaghari, Joost van de Weijer, Bartlomiej Twardowski, and Bogdan Raducanu (Aug. 2021). "Reducing Label Effort: Self-Supervised Meets Active Learning". In: *arXiv:2108.11458 [cs]*. arXiv: 2108.11458 [cs].

Bernard, Olivier, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. (2018). "Deep Learning Techniques for Automatic MRI Cardiac Multi-Structures Segmentation and Diagnosis: Is the Problem Solved?" In: *IEEE transactions on medical imaging* 37.11, pp. 2514–2525.

Bilic, Patrick, Patrick Christ, Hongwei Bran Li, Eugene Vorontsov, Avi Ben-Cohen, Georgios Kaissis, Adi Szeskin, Colin Jacobs, Gabriel Efrain Humpire Mamani, Gabriel Chartrand, et al. (2023). "The liver tumor segmentation benchmark (lits)". In: *Medical image analysis* 84, p. 102680.

Blum, Avrim and Tom Mitchell (1998). "Combining labeled and unlabeled data with co-training". In: *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pp. 92–100.

Bolelli, Federico, Luca Lumetti, Shankeeth Vinayahalingam, Mattia Di Bartolomeo, Arrigo Pellacani, Kevin Marchesini, Niels van Nistelrooij, Pieter van Lierop, Tong Xi, Yusheng Liu, Rui Xin, Tao Yang, Lisheng Wang, Haoshen Wang, Chenfan Xu, Zhiming Cui, Marek Wodzinski, Henning Müller, Yannick Kirchhoff, Maximilian R. Rokuss, Klaus Maier-Hein, Jaehwan Han, Wan Kim, Hong-Gi Ahn, Tomasz Szczepański, Michal K. Grzeszczyk, Przemyslaw Korzeniowski, Xavier Caselles Ballester Vicent amd Paolo Burgos-Artizzu, Ferran Prados Carrasco, Stefaan Berge', Bram van Ginneken, Alexandre Anesi, and Costantino Grana (Dec. 2024). "Segmenting the Inferior Alveolar Canal in CBCTs Volumes: the ToothFairy Challenge". In: *IEEE Transactions on Medical Imaging*, pp. 1–17.

Bolelli, Federico, Kevin Marchesini, Niels van Nistelrooij, Luca Lumetti, Vittorio Pipoli, Elisa Ficarra, Shankeeth Vinayahalingam, and Costantino Grana (Mar. 2025). "Segmenting Maxillofacial Structures in CBCT Volume". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Nashville, Tennessee, USA). IEEE, pp. 1–10.

Bommasani, Rishi, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. (2021). "On the opportunities and risks of foundation models". In: *arXiv preprint arXiv:2108.07258*.

Boushehri, Sayedali Shetab, Ahmad Bin Qasim, Dominik Waibel, Fabian Schmich, and Carsten Marr (Dec. 2020). *Systematic Comparison of Incomplete-Supervision Approaches for Biomedical Imaging Classification*. Preprint. Bioinformatics.

Buda, Mateusz, Atsuto Maki, and Maciej A. Mazurowski (2018). "A Systematic Study of the Class Imbalance Problem in Convolutional Neural Networks". In: *Neural Networks* 106, pp. 249–259.

Budd, Samuel, Emma C. Robinson, and Bernhard Kainz (July 2021). "A Survey on Active Learning and Human-in-the-Loop Deep Learning for Medical Image Analysis". In: *Medical Image Analysis* 71, p. 102062. arXiv: 1910.02923 [cs, eess].

Buitinck, Lars, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux (2013). "API Design for Machine Learning Software: Experiences from the Scikit-Learn Project". In: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pp. 108–122.

Burmeister, Josafat-Mattias, Marcel Fernandez Rosas, Johannes Hagemann, Jonas Kordt, Jasper Blum, Simon Shabo, Benjamin Bergner, and Christoph Lippert (July 2022). *Less Is More: A Comparison of Active Learning Strategies for 3D Medical Image Segmentation*. arXiv: 2207.00845 [cs].

Cao, Kaidi, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma (Oct. 2019). *Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss*. arXiv: 1906.07413 [cs, stat].

Caron, Mathilde, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin (Apr. 2021). "Emerging Properties in Self-Supervised Vision Transformers". In: *arXiv:2104.14294 [cs]*. arXiv: 2104.14294 [cs].

Caruana, Rich (1997). "Multitask learning". In: *Machine Learning* 28.1, pp. 41–75.

Chan, Yao-Chun, Mingchen Li, and Samet Oymak (June 2021). "On the Marginal Benefit of Active Learning: Does Self-Supervision Eat Its Cake?" In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Toronto, ON, Canada: IEEE, pp. 3455–3459.

Chapelle, Olivier, Bernhard Schölkopf, and Alexander Zien, eds. (2006). *Semi-Supervised Learning*. The MIT Press.

Chen, Liang-Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille (2017). "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs". In: *IEEE transactions on pattern analysis and machine intelligence* 40.4, pp. 834–848.

Chen, Ting, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton (June 2020). "A Simple Framework for Contrastive Learning of Visual Representations". In: *arXiv:2002.05709 [cs, stat]*. arXiv: 2002.05709 [cs, stat].

Chernoff, Herman (1959). "Sequential Design of Experiments". In: *The Annals of Mathematical Statistics* 30.3, pp. 755–770.

Çiçek, Özgün, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger (2016). "3D U-Net: learning dense volumetric segmentation from sparse annotation". In: *International conference on medical image computing and computer-assisted intervention*. Springer, pp. 424–432.

Citovsky, Gui, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar (2021). "Batch Active Learning at Scale". In: *Advances in Neural Information Processing Systems* 34, pp. 11933–11944.

Codella, Noel CF, David Gutman, M Emre Celebi, Brian Helba, Michael A Marchetti, Stephen W Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, et al. (2018). "Skin Lesion Analysis toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (Isbi), Hosted by the International Skin Imaging Collaboration (Isic)". In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. IEEE, pp. 168–172.

Colling, Pascal, Lutz Roese-Koerner, Hanno Gottschalk, and Matthias Rottmann (2020). "Metabox+: A New Region Based Active Learning Method for Semantic Segmentation Using Priority Maps". In: *arXiv preprint arXiv:2010.01884*. arXiv: 2010.01884.

Combalia, Marc, Noel CF Codella, Veronica Rotemberg, Brian Helba, Veronica Vilaplana, Ofer Reiter, Cristina Carrera, Alicia Barreiro, Allan C Halpern, Susana Puig, et al. (2019). "Bcn20000: Dermoscopic Lesions in the Wild". In: *arXiv preprint arXiv:1908.02288*. arXiv: 1908.02288.

Cordts, Marius, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele (2016a). "The Cityscapes Dataset for Semantic Urban Scene Understanding". In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

— (2016b). "The Cityscapes Dataset for Semantic Urban Scene Understanding". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3213–3223.

Cortes, Corinna and Vladimir Vapnik (1995). "Support-vector networks". In: *Machine learning* 20.3, pp. 273–297.

Cubuk, Ekin D, Barret Zoph, Jonathon Shlens, and Quoc V Le (2020). "Randaugment: Practical Automated Data Augmentation with a Reduced Search Space". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 702–703.

Czolbe, Steffen, Kasra Arnavaz, Oswin Krause, and Aasa Feragen (2021). "Is Segmentation Uncertainty Useful?" In: *Information Processing in Medical Imaging: 27th International Conference, IPMI 2021, Virtual Event, June 28–June 30, 2021, Proceedings 27*. Springer, pp. 715–726.

D. Almeida, Silvia, Tobias Norajitra, Carsten T Lüth, Tassilo Wald, Vivienn Weru, Marco Nolden, Paul F Jäger, Oyunbileg von Stackelberg, Claus Peter Heußel, Oliver Weinheimer, et al. (2024). "How do deep-learning models generalize across populations? Cross-ethnicity generalization of COPD detection". In: *Insights into imaging* 15.1, p. 198.

Dai, Jifeng, Kaiming He, and Jian Sun (2015). "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1635–1643.

Dalal, N. and B. Triggs (2005). "Histograms of oriented gradients for human detection". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1, 886–893 vol. 1.

Demšar, Janez (2006). "Statistical comparisons of classifiers over multiple data sets". In: *Journal of Machine learning research* 7.Jan, pp. 1–30.

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 248–255.

Der Kiureghian, Armen and Ove Ditlevsen (2009). "Aleatory or epistemic? Does it matter?" In: *Structural safety* 31.2, pp. 105–112.

Dice, Lee R (1945). "Measures of the Amount of Ecologic Association between Species". In: *Ecology* 26.3, pp. 297–302.

Donahue, Jeff, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell (2014). "DeCAF: A deep convolutional activation feature for generic visual recognition". In: *International Conference on Machine Learning*. PMLR, pp. 647–655.

Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby (2021). "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *International Conference on Learning Representations*.

Du, Yuxin, Fan Bai, Tiejun Huang, and Bo Zhao (2024). "Segvol: Universal and interactive volumetric medical image segmentation". In: *Advances in Neural Information Processing Systems* 37, pp. 110746–110783.

Esser, Patrick, Robin Rombach, and Bjorn Ommer (2021). "Taming transformers for high-resolution image synthesis". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883.

Fang, Meng, Yuan Li, and Trevor Cohn (Sept. 2017). "Learning How to Active Learn: A Deep Reinforcement Learning Approach". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 595–605.

Finn, Chelsea, Pieter Abbeel, and Sergey Levine (2017). "Model-agnostic meta-learning for fast adaptation of deep networks". In: *International Conference on Machine Learning*. PMLR, pp. 1126–1135.

Fisher, Ronald A (1922). "On the mathematical foundations of theoretical statistics". In: *Philosophical transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character* 222.594-604, pp. 309–368.

Föllmer, Bernhard, Kenrick Schulze, Christian Wald, Sebastian Stober, Wojciech Samek, and Marc Dewey (2024). "Active Learning with the nnUNet and Sample Selection with Uncertainty-

Aware Submodular Mutual Information Measure". In: *Submitted to Medical Imaging with Deep Learning*.

Gaillochet, Mélanie, Christian Desrosiers, and Hervé Lombaert (Dec. 2023a). "Active Learning for Medical Image Segmentation with Stochastic Batches". In: *Medical Image Analysis* 90, p. 102958.

— (Jan. 2023b). *TAAL: Test-time Augmentation for Active Learning in Medical Image Segmentation*. arXiv: 2301.06624 [cs].

Gal, Yarin and Zoubin Ghahramani (2016). "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning". In: *Proceedings of the 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, pp. 1050–1059.

Gal, Yarin, Riashat Islam, and Zoubin Ghahramani (Mar. 2017a). "Deep Bayesian Active Learning with Image Data". In: *arXiv:1703.02910 [cs, stat]*. arXiv: 1703.02910 [cs, stat].

— (2017b). "Deep Bayesian Active Learning with Image Data". In: *International Conference on Machine Learning*. PMLR, pp. 1183–1192.

Ganin, Yaroslav, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky (2016). "Domain-adversarial training of neural networks". In: *Journal of Machine Learning Research* 17.59, pp. 1–35.

Gao, Mingfei, Zizhao Zhang, Guo Yu, Sercan O. Arik, Larry S. Davis, and Tomas Pfister (July 2020). "Consistency-Based Semi-supervised Active Learning: Towards Minimizing Labeling Cost". In: *arXiv:1910.07153 [cs]*. arXiv: 1910.07153 [cs].

Gao, Mingfei, Zizhao Zhang, Guo Yu, Sercan Ö. Arık, Larry S. Davis, and Tomas Pfister (2020). "Consistency-Based Semi-supervised Active Learning: Towards Minimizing Labeling Cost". In: *Computer Vision – ECCV 2020*. Ed. by Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm. Vol. 12355. Cham: Springer International Publishing, pp. 510–526.

Garrucho, Lidia, Kaisar Kushibar, Claire-Anne Reidel, Smriti Joshi, Richard Osuala, Apostolia Tsirikoglou, Maciej Bobowicz, Javier del Riego, Alessandro Catanese, Katarzyna Gwoździewicz, Maria-Laura Cosaka, Pasant M Abo-Elhoda, Sara W Tantawy, Shorouq S Sakrana, Norhan O Shawky-Abdelfatah, Amr Muhammad Abdo Salem, Androniki Kozana, Eugen Divjak, Gordana Ivanac, Katerina Nikiforaki, Michail E Klontzas, Rosa García-Dosdá, Meltem Gulsun-Akpinar, Oğuz Lafcı, Ritse Mann, Carlos Martín-Isla, Fred Prior, Kostas Marias, Martijn P A Starmans, Fredrik Strand, Oliver Díaz, Laura Igual, and Karim Lekadir (2025). "A large-scale multicenter breast cancer DCE-MRI benchmark dataset with expert segmentations". In: *Scientific Data* 12.1, p. 453.

Geifman, Yonatan, Guy Uziel, and Ran El-Yaniv (Apr. 2019). *Bias-Reduced Uncertainty Estimation for Deep Neural Classifiers*. arXiv: 1805.08206 [cs, stat].

Geirhos, Robert, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann (Nov. 2020). "Shortcut Learning in Deep Neural Networks". In: *Nat Mach Intell* 2.11, pp. 665–673. arXiv: 2004.07780 [cs, q-bio].

Gidaris, Spyros, Praveer Singh, and Nikos Komodakis (Mar. 2018). "Unsupervised Representation Learning by Predicting Image Rotations". In: *arXiv:1803.07728 [cs]*. arXiv: 1803.07728 [cs].

Gonzalez, Camila, Karol Gotkowski, Andreas Bucher, Ricarda Fischbach, Isabel Kaltenborn, and Anirban Mukhopadhyay (2021). "Detecting When Pre-trained nnU-Net Models Fail Silently for Covid-19 Lung Lesion Segmentation". In: *Medical Image Computing and Computer Assisted Intervention MICCAI 2021*. Ed. by Marleen de Bruijne, Philippe C. Cattin, Stéphane Cotin, Nicolas Padoy, Stefanie Speidel, Yefeng Zheng, and Caroline Essert. Cham: Springer International Publishing, pp. 304–314.

Gotkowski, Karol, Carsten Lüth, Paul F Jäger, Sebastian Ziegler, Lars Krämer, Stefan Denner, Shuhan Xiao, Nico Disch, Klaus H Maier-Hein, and Fabian Isensee (2024). "Embarrassingly Simple Scribble Supervision for 3D Medical Segmentation". In: *arXiv preprint arXiv:2403.12834*. arXiv: 2403.12834.

Grandvalet, Yves and Yoshua Bengio (2004). "Semi-supervised learning by entropy minimization". In: *Advances in Neural Information Processing Systems* 17.

Grill, Jean-Bastien, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. (2020). "Bootstrap your own latent-a new approach to self-supervised learning". In: *Advances in neural information processing systems* 33, pp. 21271–21284.

Gui, Jie, Tuo Chen, Jing Zhang, Qiong Cao, Zhenan Sun, Hao Luo, and Dacheng Tao (2024). "A survey on self-supervised learning: Algorithms, applications, and future trends". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46.12, pp. 9052–9071.

Guo, Chuan, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger (2017). "On calibration of modern neural networks". In: *International conference on machine learning*. PMLR, pp. 1321–1330.

Guo, Lan-Zhe, Zhen-Yu Zhang, Yuan Jiang, Yu-Feng Li, and Zhi-Hua Zhou (2020). "Safe deep semi-supervised learning for unseen-class unlabeled data". In: *International Conference on Machine Learning*. PMLR, pp. 3897–3906.

Guo, Lan-Zhe, Zhi Zhou, and Yu-Feng Li (2022). "Robust Deep Semi-Supervised Learning: A Brief Introduction". In: *arXiv preprint arXiv:2202.05975*.

Gupte, Sanket Rajan, Josiah Aklilu, Jeffrey J. Nirschl, and Serena Yeung-Levy (Jan. 2024). *Revisiting Active Learning in the Era of Vision Foundation Models*. arXiv: 2401.14555 [cs].

Gustafsson, Fredrik K., Martin Danelljan, and Thomas B. Schön (Apr. 2020). *Evaluating Scalable Bayesian Deep Learning Methods for Robust Computer Vision*. arXiv: 1906.01620 [cs, stat].

Hancock, Matthew C. and Jerry F. Magnan (Oct. 2016). "Lung Nodule Malignancy Classification Using Only Radiologist-Quantified Image Features as Inputs to Statistical Learning Algorithms: Probing the Lung Image Database Consortium Dataset with Two Statistical Learning Methods". In: *J Med Imaging (Bellingham)* 3.4, p. 044504.

He, Kaiming, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick (2022). "Masked autoencoders are scalable vision learners". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009.

He, Kaiming, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick (Mar. 2020). "Momentum Contrast for Unsupervised Visual Representation Learning". In: *arXiv:1911.05722 [cs]*. arXiv: 1911.05722 [cs].

He, Kaiming, Ross Girshick, and Piotr Dollár (2019). "Rethinking imagenet pre-training". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4918–4927.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep Residual Learning for Image Recognition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778.

Heller, Nicholas, Fabian Isensee, Dasha Trofimova, Resha Tejpaul, Zhongchen Zhao, Huai Chen, Lisheng Wang, Alex Golts, Daniel Khapun, Daniel Shats, et al. (2023). "The Kits21 Challenge: Automatic Segmentation of Kidneys, Renal Tumors, and Renal Cysts in Corticomedullary-Phase Ct". In: *arXiv preprint arXiv:2307.01984*. arXiv: 2307.01984.

Hendrycks, Dan and Kevin Gimpel (Oct. 2018). "A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks". In: *arXiv:1610.02136 [cs]*. arXiv: 1610.02136 [cs].

Hinton, Geoffrey E., Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov (July 2012). *Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors*. arXiv: 1207.0580 [cs].

Hoeffding, Wassily (1994). "Probability Inequalities for Sums of Bounded Random Variables". In: *The Collected Works of Wassily Hoeffding*. Springer, pp. 409–426.

Holder, Christopher J. and Muhammad Shafique (Oct. 2021). "Efficient Uncertainty Estimation in Semantic Segmentation via Distillation". In: *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. Montreal, BC, Canada: IEEE, pp. 3080–3087.

Houlsby, Neil, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel (Dec. 2011). "Bayesian Active Learning for Classification and Preference Learning". In: *arXiv:1112.5745 [cs, stat]*. arXiv: 1112.5745 [cs, stat].

Hu, Edward J, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. (2022). "Lora: Low-rank adaptation of large language models." In: *ICLR* 1.2, p. 3.

Hu, Shi, Daniel Worrall, Stefan Knegt, Bas Veeling, Henkjan Huisman, and Max Welling (2019). "Supervised Uncertainty Quantification for Segmentation with Multiple Annotations". In: *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II 22*. Springer, pp. 137–145.

Hübotter, Jonas, Bhavya Sukhija, Lenart Treven, Yarden As, and Andreas Krause (Mar. 2024). *Information-Based Transductive Active Learning*. arXiv: 2402.15898 [cs].

Ioffe, Sergey and Christian Szegedy (2015). "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *International conference on machine learning*. pmlr, pp. 448–456.

Iscen, Ahmet, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum (2019). "Label propagation for deep semi-supervised learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5070–5079.

Isensee, Fabian, Paul F. Jaeger, Simon A. A. Kohl, Jens Petersen, and Klaus H. Maier-Hein (Feb. 2021a). "nnU-Net: A Self-Configuring Method for Deep Learning-Based Biomedical Image Segmentation". In: *Nat Methods* 18.2, pp. 203–211.

— (Feb. 2021b). "nnU-Net: A Self-Configuring Method for Deep Learning-Based Biomedical Image Segmentation". In: *Nat Methods* 18.2, pp. 203–211.

Isensee, Fabian, Maximilian Rokuss, Lars Krämer, Stefan Dinkelacker, Ashis Ravindran, Florian Stritzke, Benjamin Hamm, Tassilo Wald, Moritz Langenberg, Constantin Ulrich, et al. (2025). "nninteractive: Redefining 3d promptable segmentation". In: *arXiv preprint arXiv:2503.08373*.

Isensee, Fabian, Tassilo Wald, Constantin Ulrich, Michael Baumgartner, Saikat Roy, Klaus Maier-Hein, and Paul F Jaeger (2024). "Nnu-Net Revisited: A Call for Rigorous Validation in 3d Medical Image Segmentation". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 488–498.

Jaeger, Paul F, Carsten Tim Lüth, Lukas Klein, and Till J. Bungert (2023). "A Call to Reflect on Evaluation Practices for Failure Detection in Image Classification". In: *The Eleventh International Conference on Learning Representations*.

Jaster, Bjarne and Martin Kohlhase (Mar. 2025). "Trust Issues in Active Learning and Their Impact on Real-World Applications". In: *2025 IEEE Symposium on Trustworthy, Explainable and Responsible Computational Intelligence (CITREx Companion)*. Trondheim, Norway: IEEE, pp. 1–5.

Jensen, J. L. W. V. (1906). "Sur Les Fonctions Convexes et Les Inégalités Entre Les Valeurs Moyennes". In: *Acta Mathematica* 30.none, pp. 175–193.

Ji, Yilin, Daniel Kaestner, Oliver Wirth, and Christian Wressnegger (Jan. 2023). "Randomness Is the Root of All Evil: More Reliable Evaluation of Deep Active Learning". In: *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Waikoloa, HI, USA: IEEE, pp. 3932–3941.

Ji, Yuanfeng, Haotian Bai, Chongjian Ge, Jie Yang, Ye Zhu, Ruimao Zhang, Zhen Li, Lingyan Zhanng, Wanling Ma, Xiang Wan, et al. (2022). "Amos: A Large-Scale Abdominal Multi-Organ Benchmark for Versatile Medical Image Segmentation". In: *Advances in neural information processing systems* 35, pp. 36722–36732.

Joskowicz, Leo, Daphna Cohen, Noa Caplan, and Jacob Sosna (2019). "Inter-observer variability of manual contour delineation of structures in CT". In: *European Radiology* 29.3, pp. 1391–1399.

Jungo, Alain, Fabian Balsiger, and Mauricio Reyes (2020). "Analyzing the Quality and Challenges of Uncertainty Estimations for Brain Tumor Segmentation". In: *Frontiers in neuroscience*, p. 282.

Kahl, Kim-Celine, Selen Erkan, Jeremias Traub, Carsten T. Lüth, Klaus Maier-Hein, Lena Maier-hein, and Paul F Jaeger (2025). "SURE-VQA: Systematic Understanding of Robustness Evaluation in Medical VQA Tasks". In: *Transactions on Machine Learning Research*.

Kahl, Kim-Celine, Carsten T. Lüth, Maximilian Zenk, Klaus Maier-Hein, and Paul F Jaeger (2024a). "ValUES: A Framework for Systematic Validation of Uncertainty Estimation in Semantic Segmentation". In: *The Twelfth International Conference on Learning Representations*.

— (2024b). "ValUES: A Framework for Systematic Validation of Uncertainty Estimation in Semantic Segmentation". In: *The Twelfth International Conference on Learning Representations*.

Kendall, Alex and Yarin Gal (2017a). "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc.

— (2017b). "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc.

Kendall, Maurice George (1948). "Rank Correlation Methods." In.

Khoreva, Anna, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele (2017). "Simple does it: Weakly supervised instance and semantic segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 876–885.

Kidger, Patrick (2022). "On neural differential equations". In: *arXiv preprint arXiv:2202.02435*.

Kiefer, J. (1974). "General Equivalence Theory for Optimum Designs (Approximate Theory)". In: *The Annals of Statistics* 2.5, pp. 849–879.

Kim, Jaehyung, Youngbum Hur, Sejun Park, Eunho Yang, Sung Ju Hwang, and Jinwoo Shin (2020). "Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning". In: *Advances in neural information processing systems* 33, pp. 14567–14579.

Kim, Kwanyoung, Dongwon Park, Kwang In Kim, and Se Young Chun (June 2021). "Task-Aware Variational Adversarial Active Learning". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Nashville, TN, USA: IEEE, pp. 8162–8171.

Kingma, Durk P, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling (2014). "Semi-supervised learning with deep generative models". In: *Advances in Neural Information Processing Systems* 27.

Kirillov, Alexander, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. (2023). "Segment anything". In: *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026.

Kirsch, Andreas (Jan. 2024). *Advancing Deep Active Learning & Data Subset Selection: Unifying Principles with Information-Theory Intuitions*. arXiv: 2401.04305 [cs, math].

Kirsch, Andreas, Sebastian Farquhar, Parmida Atighehchian, Andrew Jesson, Frederic Branchaud-Charron, and Yarin Gal (Jan. 2022). *Stochastic Batch Acquisition for Deep Active Learning*. arXiv: 2106.12059 [cs, stat].

— (Sept. 2023). *Stochastic Batch Acquisition: A Simple Baseline for Deep Active Learning*. arXiv: 2106.12059 [cs, stat].

Kirsch, Andreas and Yarin Gal (Dec. 2021). *A Practical & Unified Notation for Information-Theoretic Quantities in ML*. arXiv: 2106.12062 [cs, stat].

— (2022). "Unifying Approaches in Active Learning and Active Sampling via Fisher Information and Information-Theoretic Quantities". In: *Transactions on Machine Learning Research*.

Kirsch, Andreas, Joost van Amersfoort, and Yarin Gal (Oct. 2019a). "BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning". In: *arXiv:1906.08158 [cs, stat]*. arXiv: 1906.08158 [cs, stat].

— (2019b). "BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc.

Klein, Lukas, Carsten Lüth, Udo Schlegel, Till Bungert, Mennatallah El-Assady, and Paul Jäger (2024). "Navigating the maze of explainable ai: A systematic approach to evaluating methods and metrics". In: *Advances in Neural Information Processing Systems* 37, pp. 67106–67146.

Kohl, Simon, Bernardino Romera-Paredes, Clemens Meyer, Jeffrey De Fauw, Joseph R. Ledsam, Klaus Maier-Hein, S. M. Ali Eslami, Danilo Jimenez Rezende, and Olaf Ronneberger (2018). "A Probabilistic U-Net for Segmentation of Ambiguous Images". In: *Advances in Neural Information Processing Systems*. Vol. 31. Curran Associates, Inc.

Kohl, Simon A. A., Bernardino Romera-Paredes, Klaus H. Maier-Hein, Danilo Jimenez Rezende, S. M. Ali Eslami, Pushmeet Kohli, Andrew Zisserman, and Olaf Ronneberger (May 2019). "A Hierarchical Probabilistic U-Net for Modeling Multi-Scale Ambiguities". In: *arXiv:1905.13077 [cs]*. arXiv: 1905.13077 [cs].

Kossen, Jannik, Sebastian Farquhar, Yarin Gal, and Tom Rainforth (June 2021). "Active Testing: Sample-Efficient Model Evaluation". In: *arXiv:2103.05331 [cs, stat]*. arXiv: 2103.05331 [cs, stat].

Kottke, Daniel, Adrian Calma, Denis Huseljic, Georg Krempl, and Bernhard Sick (n.d.). "Challenges of Reliable, Realistic and Comparable Active Learning Evaluation". In: ().

Krishnan, Ranganath, Nilesh Ahuja, Alok Sinha, Mahesh Subedar, Omesh Tickoo, and Ravi Iyer (Sept. 2021). "Improving Robustness and Efficiency in Active Learning with Contrastive Loss". In: *arXiv:2109.06873 [cs]*. arXiv: 2109.06873 [cs].

Krizhevsky, A. (2009). "Learning Multiple Layers of Features from Tiny Images". In.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton (May 2017). "ImageNet Classification with Deep Convolutional Neural Networks". In: *Commun. ACM* 60.6, pp. 84–90.

Krogh, Anders and John Hertz (1991). "A simple weight decay can improve generalization". In: *Advances in neural information processing systems* 4.

Kumar, Ananya, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang (Feb. 2022). "Fine-Tuning Can Distort Pretrained Features and Underperform Out-of-Distribution". In: *arXiv:2202.10054 [cs]*. arXiv: 2202.10054 [cs].

Laine, Samuli and Timo Aila (2017). "Temporal ensembling for semi-supervised learning". In: *International Conference on Learning Representations (ICLR)*.

Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell (2017). "Simple and scalable predictive uncertainty estimation using deep ensembles". In: *Advances in neural information processing systems* 30.

Lambert, Benjamin, Florence Forbes, Senan Doyle, Alan Tucholka, and Michel Dojat (Nov. 2022). *Improving Uncertainty-based Out-of-Distribution Detection for Medical Image Segmentation*. arXiv: 2211.05421 [cs, eess].

Laradji, Issam H, Pau Rodriguez, Nazanin Mohammadi Kani, Abhishek Sharma, Keenan Lensink, Derek Jacobus Law, Svetlana Popescu, Derek Nowrouzezahrai, and David Vazquez (2021). "Weakly supervised segmentation with point annotations for histopathology images via contrast-based variational model". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3112–3121.

LeCun, Y., B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel (Dec. 1989). "Backpropagation Applied to Handwritten Zip Code Recognition". In: *Neural Computation* 1.4, pp. 541–551.

LeCun, Yann (1998). "The MNIST Database of Handwritten Digits". In: *http://yann. lecun. com/exdb/mnist/*.

Lee, Dong-Hyun et al. (2013). "Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks". In: *Workshop on Challenges in Representation Learning, ICML*. Vol. 3, p. 896.

Lee, Jungbeom, Jooyoung Yi, Chaehun Shin, and Sungroh Yoon (2021). "Railroad is not a train: Saliency as pseudo-pixel supervision for weakly supervised semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5495–5505.

Lehmann, Erich Leo and George Casella (1998). *Theory of point estimation*. Springer.

Li, Zihan, Yuan Zheng, Dandan Shan, Shuzhou Yang, Qingde Li, Beizhan Wang, Yuanting Zhang, Qingqi Hong, and Dinggang Shen (2024). "Scribformer: Transformer makes cnn work better for scribble-based medical image segmentation". In: *IEEE Transactions on Medical Imaging* 43.6, pp. 2254–2265.

Lin, Di, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun (2016). "Scribblesup: Scribble-supervised convolutional networks for semantic segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3159–3167.

Lin, Guosheng, Chunhua Shen, Anton Van Den Hengel, and Ian Reid (2016). "Efficient piecewise training of deep structured models for semantic segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3194–3203.

Litjens, Geert, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez (2017). "A survey on deep learning in medical image analysis". In: *Medical image analysis* 42, pp. 60–88.

Liu, Peng, Lizhe Wang, Guojin He, and Lei Zhao (Feb. 2022). *A Survey on Active Deep Learning: From Model-driven to Data-driven*. arXiv: 2101.09933 [cs].

Liu, Xiao, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang (2021). "Self-Supervised Learning: Generative or Contrastive". In: *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1.

Long, Jonathan, Evan Shelhamer, and Trevor Darrell (2015). "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440.

Long, Mingsheng, Yue Cao, Jianmin Wang, and Michael Jordan (2015). "Learning transferable features with deep adaptation networks". In: *International Conference on Machine Learning*. PMLR, pp. 97–105.

Lowe, David G (2004). "Distinctive image features from scale-invariant keypoints". In: *International journal of computer vision* 60.2, pp. 91–110.

Lumetti, Luca, Vittorio Pipoli, Federico Bolelli, Elisa Ficarra, and Costantino Grana (2024). "Enhancing Patch-Based Learning for the Segmentation of the Mandibular Canal". In: *IEEE Access*, pp. 1–12.

Luo, Xiangde, Wenjun Liao, Jianghong Xiao, Jieneng Chen, Tao Song, Xiaofan Zhang, Kang Li, Dimitris N Metaxas, Guotai Wang, and Shaoting Zhang (2022). "WORD: A large scale dataset, benchmark and clinical applicable study for abdominal organ segmentation from CT image". In: *Medical Image Analysis* 82, p. 102642.

Luo, Zhiming, Frédéric Branchaud-Charron, Carl Lemaire, Janusz Konrad, Shaozi Li, Akshaya Mishra, Andrew Achkar, Justin Eichel, and Pierre-Marc Jodoin (Oct. 2018). "MIO-TCD: A New Benchmark Dataset for Vehicle Classification and Localization". In: *IEEE Trans. on Image Process.* 27.10, pp. 5129–5141.

Lüth, Carsten T., Jeremias Traub, Kim-Celine Kahl, Till J. Bungert, Lukas Klein, Lars Krämer, Paul F. Jaeger, Fabian Isensee, and Klaus Maier-Hein (2025). "nnActive: A Framework for Evaluation of Active Learning in 3D Biomedical Segmentation". In: *Transactions on Machine Learning Research*.

Lüth, Carsten T., Jeremias Traub, Kim-Celine Kahl, Till J. Bungert, Lukas Klein, Lars Krämer, Paul F. Jaeger, Klaus Maier-Hein, and Fabian Isensee (2025). "Finally outshining the Random Baseline: A simple and effective solution for Active Learning in 3D biomedical imaging". In: *Submitted to Transactions on Machine Learning Research*. Under review.

Lüth, Carsten T., David Zimmerer, Gregor Koehler, Paul F. Jaeger, Fabian Isensee, Jens Petersen, and Klaus H. Maier-Hein (Jan. 2023). *CRADL: Contrastive Representations for Unsupervised Anomaly Detection and Localization*. arXiv: 2301.02126 [cs].

Lüth, Carsten Tim, Till J. Bungert, Lukas Klein, and Paul F Jaeger (2023). "Navigating the Pitfalls of Active Learning Evaluation: A Systematic Framework for Meaningful Performance Assessment". In: *Thirty-Seventh Conference on Neural Information Processing Systems*.

Ma, Jun, Yuting He, Feifei Li, Lin Han, Chenyu You, and Bo Wang (2024). "Segment anything in medical images". In: *Nature Communications* 15.1, p. 654.

Ma, Jun, Feifei Li, and Bo Wang (2024). "U-mamba: Enhancing long-range dependency for biomedical image segmentation". In: *arXiv preprint arXiv:2401.04722*.

Ma, Siteng, Haochang Wu, Aonghus Lawlor, and Ruihai Dong (Jan. 2024). *Breaking the Barrier: Selective Uncertainty-based Active Learning for Medical Image Segmentation*. arXiv: 2401.16298 [cs].

Mackowiak, Radek, Philip Lenz, Omair Ghori, Ferran Diego, Oliver Lange, and Carsten Rother (Oct. 2018). "CEREALS - Cost-Effective REgion-based Active Learning for Semantic Segmentation". In: *BMVC*. arXiv: 1810.09726.

Maier-Hein, Lena, Annika Reinke, Patrick Godau, Minu D Tizabi, Florian Buettner, Evangelia Christodoulou, Ben Glocker, Fabian Isensee, Jens Kleesiek, Michal Kozubek, et al. (2024). "Metrics reloaded: recommendations for image analysis validation". In: *Nature methods* 21.2, pp. 195–212.

Mehrtash, Alireza, William M Wells, Clare M Tempany, Purang Abolmaesumi, and Tina Kapur (2020). "Confidence Calibration and Predictive Uncertainty Estimation for Deep Medical Image Segmentation". In: *IEEE transactions on medical imaging* 39.12, pp. 3868–3878.

Mehta, Raghav, Angelos Filos, Yarin Gal, and Tal Arbel (May 2020). *Uncertainty Evaluation Metric for Brain Tumour Segmentation*. arXiv: 2005.14262 [cs, eess].

Menze, Bjoern H, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. (2014). "The multimodal brain tumor image segmentation benchmark (BRATS)". In: *IEEE transactions on medical imaging* 34.10, pp. 1993–2024.

Mittal, Sudhanshu, J. Niemeijer, J. Schäfer, and Thomas Brox (2023). "Best Practices in Active Learning for Semantic Segmentation". In: *German Conference on Pattern Recognition (GCPR)*.

Mittal, Sudhanshu, Joshua Niemeijer, Jörg P. Schäfer, and Thomas Brox (Mar. 2023). *Best Practices in Active Learning for Semantic Segmentation*. arXiv: 2302.04075 [cs].

Mittal, Sudhanshu, Maxim Tatarchenko, Özgün Çiçek, and Thomas Brox (Dec. 2019a). *Parting with Illusions about Deep Active Learning*. arXiv: 1912.05361 [cs].

— (Dec. 2019b). "Parting with Illusions about Deep Active Learning". In: *arXiv:1912.05361 [cs]*. arXiv: 1912.05361 [cs].

Miyato, Takeru, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii (2018). "Virtual adversarial training: a regularization method for supervised and semi-supervised learning". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.8, pp. 1979–1993.

Mobiny, Aryan, Pengyu Yuan, Supratik K Moulik, Naveen Garg, Carol C Wu, and Hien Van Nguyen (2021). "Dropconnect Is Effective in Modeling Uncertainty of Bayesian Deep Networks". In: *Scientific reports* 11.1, pp. 1–14.

Monteiro, Miguel, Loic Le Folgoc, Daniel Coelho de Castro, Nick Pawlowski, Bernardo Marques, Konstantinos Kamnitsas, Mark van der Wilk, and Ben Glocker (2020). "Stochastic Segmentation Networks: Modelling Spatially Correlated Aleatoric Uncertainty". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., pp. 12756–12767.

Mukhoti, Jishnu, Andreas Kirsch, Joost van Amersfoort, Philip H. S. Torr, and Yarin Gal (Jan. 2022). *Deep Deterministic Uncertainty: A Simple Baseline*. arXiv: 2102.11582 [cs, stat].

Mukhoti, Jishnu, Joost van Amersfoort, Philip HS Torr, and Yarin Gal (2021). "Deep Deterministic Uncertainty for Semantic Segmentation". In: *arXiv preprint arXiv:2111.00079*. arXiv: 2111.00079.

Munjal, Prateek, Nasir Hayat, Munawar Hayat, Jamshid Sourati, and Shadab Khan (Apr. 2022a). "Towards Robust and Reproducible Active Learning Using Neural Networks". In: *arXiv:2002.09564 [cs, stat]*. arXiv: 2002.09564 [cs, stat].

— (2022b). "Towards Robust and Reproducible Active Learning Using Neural Networks". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 223–232.

Nash, Will, Liang Zheng, and Nick Birbilis (Mar. 2022). "Deep Learning Corrosion Detection with Confidence". In: *npj Materials Degradation* 6.1, p. 26.

Nath, Vishwesh, Dong Yang, Bennett A. Landman, Daguang Xu, and Holger R. Roth (Oct. 2021). "Diminishing Uncertainty within the Training Pool: Active Learning for Medical Image Segmentation". In: *IEEE Trans. Med. Imaging* 40.10, pp. 2534–2547. arXiv: 2101.02323 [cs].

Nemenyi, P. (1963). "Distribution-free multiple comparisons". In: *PhD Thesis, Princeton University*.

Neumann, Lukas, Andrew Zisserman, and Andrea Vedaldi (n.d.). "Relaxed Softmax: Efficient Confidence Auto-Calibration for Safe Pedestrian Detection". In: ().

Neyshabur, Behnam, Hanie Sedghi, and Chiyuan Zhang (2020). "What is being transferred in transfer learning?" In: *Advances in Neural Information Processing Systems* 33, pp. 512–523.

Noroozi, Mehdi and Paolo Favaro (Aug. 2017). "Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles". In: *arXiv:1603.09246 [cs]*. arXiv: 1603.09246 [cs].

Oliver, Avital, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian J. Goodfellow (June 2019). "Realistic Evaluation of Deep Semi-Supervised Learning Algorithms". In: *arXiv:1804.09170 [cs, stat]*. arXiv: 1804.09170 [cs, stat].

Oquab, Maxime, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski (2024). "DINOv2: Learning Robust Visual Features without Supervision". In: *Transactions on Machine Learning Research*. Featured Certification.

Pan, Sinno Jialin and Qiang Yang (2009). "A survey on transfer learning". In: *IEEE Transactions on Knowledge and Data Engineering* 22.10, pp. 1345–1359.

Papandreou, George, Liang-Chieh Chen, Kevin P Murphy, and Alan L Yuille (2015). "Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1742–1750.

Pathak, Deepak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros (Nov. 2016). "Context Encoders: Feature Learning by Inpainting". In: *arXiv:1604.07379 [cs]*. arXiv: 1604.07379 [cs].

Peirce, Charles S (1883). "A theory of probable inference." In.

Peirce, Charles Sanders (1877). "Illustrations of the Logic of Science". In: *Popular Science Monthly* 12–13. Series of articles published 1877–1878.

— (Nov. 1882). "Introductory Lecture on the Study of Logic". In: *Johns Hopkins University Circulars* 2.19. Lecture delivered September 1882. Reprinted in *Collected Papers of Charles Sanders Peirce*, vol. 7, pars. 59–76; *Writings of Charles S. Peirce*, vol. 4, pp. 378–382; and *The Essential Peirce*, vol. 1, pp. 210–214., pp. 11–12.

Pinsler, Robert, Jonathan Gordon, Eric Nalisnick, and José Miguel Hernández-Lobato (Feb. 2021). "Bayesian Batch Active Learning as Sparse Subset Approximation". In: *arXiv:1908.02144 [cs, stat]*. arXiv: 1908.02144 [cs, stat].

Postels, Janis, Francesco Ferroni, Huseyin Coskun, Nassir Navab, and Federico Tombari (2019). "Sampling-Free Epistemic Uncertainty Estimation Using Approximated Variance Propagation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2931–2940.

Postels, Janis, Mattia Segu, Tao Sun, Luc Van Gool, Fisher Yu, and Federico Tombari (2021). "On the Practicality of Deterministic Epistemic Uncertainty". In: *arXiv preprint arXiv:2107.00649*. arXiv: 2107.00649.

Qiao, Siyuan, Wei Shen, Zhishuai Zhang, Bo Wang, and Alan Yuille (2018). "Deep co-training for semi-supervised image recognition". In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 135–152.

Radford, Alec, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. (2021). "Learning transferable visual models from natural language supervision". In: pp. 8748–8763.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. (2019). "Language models are unsupervised multitask learners". In: *OpenAI blog* 1.8, p. 9.

Richter, Stephan R., Vibhav Vineet, Stefan Roth, and Vladlen Koltun (2016). "Playing for Data: Ground Truth from Computer Games". In: *European Conference on Computer Vision (ECCV)*. Ed. by Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling. Vol. 9906. LNCS. Springer International Publishing, pp. 102–118.

Ridnik, Tal, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor (2021). "Imagenet-21k pre-training for the masses". In: *arXiv preprint arXiv:2104.10972*.

Robbins, Herbert (1952). "Some Aspects of the Sequential Design of Experiments". In: *Bulletin of the American Mathematical Society* 58.5, pp. 527–535.

Rodrigues, Filipe and Francisco Pereira (2018). "Deep learning from crowds". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32.1.

Rombach, Robin, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer (2022). "High-resolution image synthesis with latent diffusion models". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695.

Romberg, Julia, Christopher Schröder, Julius Gonsior, Katrin Tomanek, and Fredrik Olsson (Mar. 2025). *Have LLMs Made Active Learning Obsolete? Surveying the NLP Community*. arXiv: 2503.09701 [cs].

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (May 2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *arXiv:1505.04597 [cs]*. arXiv: 1505.04597 [cs].

Roy, Saikat, Gregor Koehler, Constantin Ulrich, Michael Baumgartner, Jens Petersen, Fabian Isensee, Paul F Jaeger, and Klaus H Maier-Hein (2023). "Mednext: Transformer-Driven Scaling of Convnets for Medical Image Segmentation". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 405–415.

Ruder, Sebastian (2017). "An overview of multi-task learning in deep neural networks". In: *arXiv preprint arXiv:1706.05098*.

Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei (2015a). "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3, pp. 211–252.

— (2015b). "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3, pp. 211–252.

Salimans, Tim, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen (2016). "Improved techniques for training gans". In: *Advances in Neural Information Processing Systems* 29.

Sener, Ozan and Silvio Savarese (2018a). "Active Learning for Convolutional Neural Networks: A Core-Set Approach". In: *International Conference on Learning Representations*.

— (June 2018b). "Active Learning for Convolutional Neural Networks: A Core-Set Approach". In: *arXiv:1708.00489 [cs, stat]*. arXiv: 1708.00489 [cs, stat].

Settles, Burr (2009). *Active Learning Literature Survey*. Computer Sciences Technical Report 1648. University of Wisconsin–Madison.

— (May 2011). "From Theories to Queries: Active Learning in Practice". In: *Active Learning and Experimental Design Workshop in Conjunction with AISTATS 2010*. Ed. by Isabelle Guyon,

Gavin Cawley, Gideon Dror, Vincent Lemaire, and Alexander Statnikov. Vol. 16. Proceedings of Machine Learning Research. Sardinia, Italy: PMLR, pp. 1–18.

Shannon, C. E. (July 1948). "A Mathematical Theory of Communication". In: *Bell System Technical Journal* 27.3, pp. 379–423.

Sharif Razavian, Ali, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson (2014). "CNN features off-the-shelf: an astounding baseline for recognition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 806–813.

Shi, Jun, Shulan Ruan, Ziqi Zhu, Minfan Zhao, Hong An, Xudong Xue, and Bing Yan (Aug. 2024). "Predictive Accuracy-Based Active Learning for Medical Image Segmentation". In: *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*. International Joint Conferences on Artificial Intelligence Organization, pp. 4885–4893.

Shi, Yuyan, Jialu Ma, Jin Yang, Shasha Wang, and Yichi Zhang (2024). "Beyond pixel-wise supervision for medical image segmentation: From traditional models to foundation models". In: *arXiv preprint arXiv:2404.13239*.

Shimizu, Atsushi, Xiaoou Cheng, Christopher Musco, and Jonathan Weare (May 2024). *Improved Active Learning via Dependent Leverage Score Sampling*. arXiv: `2310.04966 [cs]`.

Shotton, Jamie, John Winn, Carsten Rother, and Antonio Criminisi (2006). "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation". In: *European conference on computer vision*. Springer, pp. 1–15.

Siméoni, Oriane, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. (2025). "Dinov3". In: *arXiv preprint arXiv:2508.10104*.

Sinha, Samrath, Sayna Ebrahimi, and Trevor Darrell (Oct. 2019). "Variational Adversarial Active Learning". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. Seoul, Korea (South): IEEE, pp. 5971–5980.

Snell, Jake, Kevin Swersky, and Richard Zemel (2017). "Prototypical networks for few-shot learning". In: *Advances in Neural Information Processing Systems* 30.

Sohn, Kihyuk, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li (2020). "FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., pp. 596–608.

Sohn, Kihyuk, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel (Jan. 2020). "FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence". In: *arXiv:2001.07685 [cs, stat]*. arXiv: `2001.07685 [cs, stat]`.

Song, Chunfeng, Yan Huang, Wanli Ouyang, and Liang Wang (2019). "Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3136–3145.

Srinivas, Niranjan, Andreas Krause, Sham Kakade, and Matthias Seeger (2010). "Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design". In: *Proceedings of the 27th International Conference on Machine Learning*. ICML'10. Haifa, Israel: Omnipress, pp. 1015–1022.

Tajbakhsh, Nima, Laura Jeyaseelan, Qian Li, Jeffrey N Chiang, Zhihao Wu, and Xiaowei Ding (2020). "Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation". In: *Medical image analysis* 63, p. 101693.

Tang, Meng, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers (2018). "Normalized cut loss for weakly-supervised CNN segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1818–1827.

Tanno, Ryutaro, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C Alexander, and Nathan Silberman (2019). "Learning from noisy labels by regularized estimation of annotator confusion". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11244–11253.

Tomanek, Katrin and Fredrik Olsson (June 2009). "A Web Survey on the Use of Active Learning to Support Annotation of Text Data". In: *Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*. Ed. by Eric Ringger, Robbie Haertel, and Katrin Tomanek. Boulder, Colorado: Association for Computational Linguistics, pp. 45–48.

Torralba, A., P. Isola, and W.T. Freeman (2024). *Foundations of Computer Vision*. Adaptive Computation and Machine Learning series. MIT Press.

Traub, Jeremias, Till J. Bungert, Carsten T. Lüth, Michael Baumgartner, Klaus H. Maier-Hein, Lena Maier-Hein, and Paul F. Jäger (2024). "Overcoming Common Flaws in the Evaluation of Selective Classification Systems". In: *Advances in Neural Information Processing Systems*. Ed. by A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang. Vol. 37. Curran Associates, Inc., pp. 2323–2347.

Tschandl, Philipp, Cliff Rosendahl, and Harald Kittler (Dec. 2018). "The HAM10000 Dataset, a Large Collection of Multi-Source Dermatoscopic Images of Common Pigmented Skin Lesions". In: *Sci Data* 5.1, p. 180161.

Tseng, Chiung-Yi, Junhao Song, Ziqian Bi, Tianyang Wang, Chia Xin Liang, and Ming Liu (Apr. 2025). *Active Learning Methods for Efficient Data Utilization and Model Performance Enhancement*. arXiv: 2504.16136 [cs].

Tzeng, Eric, Judy Hoffman, Kate Saenko, and Trevor Darrell (2017). "Adversarial discriminative domain adaptation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7167–7176.

Van Engelen, Jesper E and Holger H Hoos (2020). "A survey on semi-supervised learning". In: *Machine Learning* 109.2, pp. 373–440.

Vepa, Arvind Murari, ZUKANG YANG, Andrew Choi, Jungseock Joo, Fabien Scalzo, and Yizhou Sun (2024). "Integrating Deep Metric Learning with Coreset for Active Learning in 3D Segmentation". In: *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*.

Vernaza, Paul and Manmohan Chandraker (2017). "Learning random-walk label propagation for weakly-supervised semantic segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7158–7166.

Wald, Abraham (1947). *Sequential Analysis*. Reprinted by Dover Publications, 1973. New York: John Wiley & Sons.

Wald, Tassilo, Saikat Roy, Fabian Isensee, Constantin Ulrich, Sebastian Ziegler, Dasha Trofimova, Raphael Stock, Michael Baumgartner, Gregor Köhler, and Klaus Maier-Hein (2025). "Primus: Enforcing attention usage for 3d medical image segmentation". In: *arXiv preprint arXiv:2503.01835*.

Wang, Guotai, Wenqi Li, Michael Aertsen, Jan Deprest, Sebastien Ourselin, and Tom Vercauteren (Apr. 2019). "Aleatoric Uncertainty Estimation with Test-Time Augmentation for Medical Image Segmentation with Convolutional Neural Networks". In: *Neurocomputing* 338, pp. 34–45. arXiv: 1807.07356.

Wang, Haoran, Qiuye Jin, Shiman Li, Siyu Liu, Manning Wang, and Zhijian Song (July 2024). "A Comprehensive Survey on Deep Active Learning in Medical Image Analysis". In: *Medical Image Analysis* 95, p. 103201.

Wang, Jingdong, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao (2020). "Deep High-Resolution Representation Learning for Visual Recognition". In: *TPAMI*.

Wang, Ruize, Duyu Tang, Nan Duan, Zhongyu Wei, Xuan-Jing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou (2021). "K-adapter: Infusing knowledge into pre-trained models with adapters". In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 1405–1418.

Wang, Yidong, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, et al. (2022). "Freematch: Self-adaptive thresholding for semi-supervised learning". In: *International Conference on Learning Representations (ICLR)*.

Wang, Yude, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen (2020). "Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12275–12284.

Wang, Zifu, Maxim Berman, Amal Rannen-Triki, Philip Torr, Devis Tuia, Tinne Tuytelaars, Luc V Gool, Jiaqian Yu, and Matthew Blaschko (2023). "Revisiting evaluation metrics for semantic segmentation: Optimization and evaluation of fine-grained intersection over union". In: *Advances in Neural Information Processing Systems* 36, pp. 60144–60225.

Weiss, Karl, Taghi M Khoshgoftaar, and DingDing Wang (2016). "A survey of transfer learning". In: *Journal of Big Data* 3.1, pp. 1–40.

Whitbread, Luke and Mark Jenkinson (2022). "Uncertainty Categories in Medical Image Segmentation: A Study of Source-Related Diversity". In: *Uncertainty for Safe Utilization of Machine*

Learning in Medical Imaging: 4th International Workshop, UNSURE 2022, Held in Conjunction with MICCAI 2022, Singapore, September 18, 2022, Proceedings. Springer, pp. 26–35.

Xie, Binhui, Longhui Yuan, Shuang Li, Chi Harold Liu, and Xinjing Cheng (2022). "Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8068–8078.

Xie, Enze, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo (2021). "SegFormer: Simple and efficient design for semantic segmentation with transformers". In: *Advances in neural information processing systems* 34, pp. 12077–12090.

Xie, Qizhe, Minh-Thang Luong, Eduard Hovy, and Quoc V Le (2020). "Self-training with noisy student improves imagenet classification". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10687–10698.

Xie, Shuai, Zunlei Feng, Ying Chen, Songtao Sun, Chao Ma, and Mingli Song (2020). "Deal: Difficulty-aware Active Learning for Semantic Segmentation". In: *Proceedings of the Asian Conference on Computer Vision*.

Yi, John Seon Keun, Minseok Seo, Jongchan Park, and Dong-Geol Choi (Jan. 2022). "Using Self-Supervised Pretext Tasks for Active Learning". In: *arXiv:2201.07459 [cs]*. arXiv: `2201.07459 [cs]`.

Yoo, Donggeun and In So Kweon (June 2019a). "Learning Loss for Active Learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

— (June 2019b). "Learning Loss for Active Learning". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA: IEEE, pp. 93–102.

Yosinski, Jason, Jeff Clune, Yoshua Bengio, and Hod Lipson (2014). "How transferable are features in deep neural networks?" In: *Advances in Neural Information Processing Systems* 27.

Zenk, Maximilian, David Zimmerer, Fabian Isensee, Paul F Jäger, Jakob Wasserthal, and Klaus Maier-Hein (2022). "Realistic Evaluation of FixMatch on Imbalanced Medical Image Classification Tasks". In: *Bildverarbeitung Für Die Medizin 2022*. Springer, pp. 291–296.

Zhan, Xueying, Huan Liu, Qing Li, and Antoni B Chan (2021). "A Comparative Survey: Benchmarking for Pool-Based Active Learning." In: *IJCAI*, pp. 4679–4686.

Zhan, Xueying, Qingzhong Wang, Kuan-hao Huang, Haoyi Xiong, Dejing Dou, and Antoni B Chan (2022a). "A Comparative Survey of Deep Active Learning". In: *arXiv preprint arXiv:2203.13450*. arXiv: `2203.13450`.

— (May 2022b). *A Comparative Survey of Deep Active Learning*. arXiv: `2203.13450 [cs]`.

Zhang, Bowen, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki (2021). "Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling". In: *Advances in Neural Information Processing Systems*. Vol. 34, pp. 18408–18419.

Zhang, Bowen, Yidong Yao, Guosheng Li, and Yu Qiao (2020). "Weakly supervised semantic segmentation for social images". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2718–2727.

Zhang, Ge, Hao Dang, and Yulong Xu (Feb. 2022). "Epistemic and Aleatoric Uncertainties Reduction with Rotation Variation for Medical Image Segmentation with ConvNets". In: *SN Appl. Sci.* 4.2, p. 56.

Zhang, Richard, Phillip Isola, and Alexei A. Efros (Oct. 2016). "Colorful Image Colorization". In: *arXiv:1603.08511 [cs]*. arXiv: `1603.08511 [cs]`.

Zhang, Tao, Guosheng Lin, Jianfei Cai, Tianyu Shen, Chunhua Shen, and Alex C Kot (2021). "A survey on weakly supervised semantic segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.9, pp. 4826–4841.

Zhou, Bolei, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba (2016). "Learning deep features for discriminative localization". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2921–2929.

Zhou, Zhi-Hua (2018). "A brief introduction to weakly supervised learning". In: *National Science Review* 5.1, pp. 44–53.

Zhu, Xiaojin, Zoubin Ghahramani, and John D Lafferty (2003). "Semi-supervised learning using gaussian fields and harmonic functions". In: *Proceedings of the 20th International Conference on Machine Learning (ICML)*, pp. 912–919.

Appendix A

Principled Evaluation of Active Learning for Image Classification

This chapter of the Appendix uses the Appendix of Carsten Tim Lüth, Till J. Bungert, Lukas Klein, and Paul F Jaeger (2023). "Navigating the Pitfalls of Active Learning Evaluation: A Systematic Framework for Meaningful Performance Assessment". In: *Thirty-Seventh Conference on Neural Information Processing Systems*.

A.1 Active learning literature, in more detail

We will discuss the current literature landscape of deep active classification with a focus on our proposed key-pitfalls as shown in fig. 5.1b.

The rules for evaluation of each of the five pitfalls (P1-P5) are:

P1 Data distribution	Use of multiple datasets for evaluation featuring class-imbalanced datasets.
P2 Starting budget	Evaluation or ablating the influence of the starting budget on multiple datasets explicitly.
P3 Query size	Evaluation or ablating the influence of the query size on multiple datasets explicitly.
P4 Performant baselines	Performance is close to ours or Munjal et al. (2022a) for ST models on CIFAR-10/100(see section A.7 for details).[1]
P4 HP Optim. & Val. Split	The use of a dedicated validation set to configure the classifier.
P5 Self-SL	Benchmarking AL with a performant Self-SL training paradigms.
P5 Semi-SL	Benchmarking AL with a performant Semi-SL training paradigm

Munjal et al. (2022a) Evaluate the performance of AL methods and compare against and with well finetuned baseline models using AutoML.

P1 Data Distribution: Perform experiments on CIFAR-10, CIFAR-100 and limited experiments on ImageNet. They perform an ablation on CIFAR-100 with an artificial imbalanced dataset.
→ (✔) due to limited imbalanced datasets.

P2 Starting Budget: Perform no experiments at all regarding the starting budget.
→ **X**

P3 Query Size: Perform ablations on CIFAR-10/100 comparing query sizes of 5% (2500) to 10% (5000). → ✔

P4 Performant Baselines: They achieve performance on CIFAR-10/100 on par with ours (see section A.7). → ✔

P4 HP Optim. & Val. Split They explicitly use a validation set and finetune their hyperparameters based on the validation set performance using AutoML.
→ ✔

P5 Self-SL: They do not consider using models pre-trained with Self-SL.
→ **X**

P5 Semi-SL: They do not consider using models trained with semi-supervised training paradigms.
→ **X**

Mittal, Tatarchenko, et al. (2019a) Evaluate the performance of AL methods and set them into context with semi-supervised training paradigms.

P1 Data Distribution: Perform experiments on CIFAR-10, CIFAR-100.
→ **X**

P2 Starting Budget: Perform experiments both on the standard setting with starting budget of 5000 (10%) on CIFAR-10/100 as well as 250 (CIFAR-10) and 500 (CIFAR-100).
→ (✔) due to limited settings.

P3 Query Size: They do not consider speicifally ablating the query size.
→ **X**

P4 Performant Baselines: Their random baseline is more performant on CIFAR10/100 than most of the literature. However, not as good as Munjal et al. (2022a) or ours (see section A.7).

[1] We only base this on ST models, as getting good performance with Self-SL and Semi-SL for low data settings can be achieved without taking HP configuration into account, as they can simply be taken from a paper focusing on them which often use very large validation sets.

→ (✔) due to performance being good but no on par with ours.

P4 HP Optim. & Val. Split They do not state optimizing their hyperparameters based on a validation set.
→ **X**

P5 Self-SL: They do not consider using models pre-trained with Self-SL.
→ **X**

P5 Semi-SL: They evaluate AL with and against the semi-supervised training paradigm 'Unsupervised Data Augmentation for Consistency Training'.
→ ✔

Bengar et al. (2021) Evaluate the performance of AL methods and set them into context with self-supervised training paradigms.

P1 Data Distribution: They perform experiments on CIFAR-10/100 and TinyImageNet. All of which are class balanced datasets.
→ **X** due to only evaluating balanced datasets.

P2 Starting Budget: They use 3 different starting budgets on each of their three datasets. CIFAR-10: 0.1%, 1%, 10%; CIFAR-100: 1%, 2%, 10%; Tiny ImageNet: 1%, 2%, 10% (% of the whole dataset).
→ ✔

P3 Query Size: Each of the three different starting budgets has a different query size resulting in overlapping experiments. Therefore it would be possible to draw some conclusions about the influence of the query size.
→ (✔) due missing selective evaluation of query size.

P4 Performant Baselines: Their supervised random baseline is performing worse on CIFAR-10/100 than most models in the literature (see section A.7).
→ **X**

P4 HP Optim. & Val. Split: They do not state optimizing their hyperparameters based on a validation set.

P5 Self-SL: They evaluate AL methods with and against one self-supervised training paradigm (SimSiam).
→ ✔

P5 Semi-Supervise Learning: They do not consider using models trained with semi-supervised training paradigms.
→ **X**

Gao, Z. Zhang, Yu, Arık, et al. (2020) Evaluate the performance of AL methods against and in the context of semi-supervised training paradigms. Further, they propose a new query method designed for AL with models that are trained with a Semi-SL training paradigm.

P1 Data Distribution: They perform experiments on CIFAR-10/100 and ImageNet.
→ **X** due to only evaluating balanced datasets.

P2 Starting Budget: They perform a specific ablation about the importance of the starting budget on CIFAR-10 with multiple settings and discuss it.
→ ✔

P3 Query Size: In addition to the standard experiments, they perform experiments with query sizes of 50 and 250. However, they do not specifically discuss its importance.
→ (✔) due missing selective evaluation of query size.

P4 Performant Baselines: The performance of their supervised random baseline models in the main comparison is not close to our performance on CIFAR-10/100.
→ **X**

P4 HP Optim. & Val. Split: They do not state optimizing their hyperparameters based on a

validation set.
→ **X**

P5 Self-SL: They do not consider using models pre-trained with Self-SL.
→ **X**

P5 Semi-SL: They evaluate AL with and against the semi-supervised training paradigm (MixMatch).
→ ✔

J. S. K. Yi et al. (2022) They propose to use self-supervised pre-text as a basis for query functions.

P1 Data Distribution: Perform experiments on CIFAR-10, an imbalanced version of CIFAR-10, Caltech-101 and ImageNet.
→ ✔

P2 Starting Budget: One experiment is performed where they select the starting budget with their proposed Active Learning method on CIFAR-10. Otherwise, they do not evaluate the performance under different starting budgets.
→ **X**

P3 Query Size: They do not evaluate the performance with regard to different query sizes.
→ **X**

P4 Performant Baselines: Their supervised random baseline models are not close to the performance of our random baseline models on CIFAR-10 (see section A.7).
→ **X**

P4 HP Optim. & Val. Split: They do not state optimizing their hyperparameters based on a validation set.
→ **X**

P5 Self-SL: They consider several different Self-SL paradigms ('Rotation Prediction', 'Colorization', 'Solving jigsaw puzzles' and 'SimSiam') based on which they select rotation prediction for their experiments.
→ (✔) due to selection of non state-of-the-art Self-SL paradigm.

P5 Semi-Supervise Learning: They do not consider using models trained with semi-supervised training paradigms.
→ **X**

Krishnan et al. (2021) They propose to use a supervised contrastive training paradigm as a basis for two AL methods.

P1 Data Distribution: Perform experiments on Fashion-MNIST, SVHN and CIFAR-10 and an imbalanced version of CIFAR-10.
→ (✔) due to imbalanced CIFAR-10 being simulated.

P2 Starting Budget: They do not evaluate the performance under different starting budgets.
→ **X**

P3 Query Size: They do not evaluate the performance with regard to different query sizes.
→ **X**

P4 Performant Baselines: Their supervised random baseline models are not close to the performance of our random baseline models on CIFAR-10 (see section A.7).
→ **X**

P4 HP Optim. & Val. Split: They do not state optimizing their hyperparameters based on a validation set.
→ **X**

P5 Self-SL: They do not consider using models pre-trained with Self-SL.
→ **X**

P5 Semi-Supervise Learning: They do not consider using models trained with semi-supervised training paradigms.
→ **X**

K. Kim et al. (2021) They propose task-aware active learning which is a combination of learning loss active learning and variational adversarial active learning.

P1 Data Distribution: Perform experiments on CIFAR-10, CIFAR-100, CALTECH 101 and imbalanced CIFARS.
→ ✔

P2 Starting Budget: They do not evaluate the performance under different starting budgets.
→ (✔)

P3 Query Size: They do not evaluate the performance with regard to different query sizes.
→ (✔)

P4 Performant Baselines: Their supervised random baseline models perform good but not on par with ours on CIFAR-10 and 100.
→ (✔)

P4 HP Optim. & Val. Split: They do not state optimizing their hyperparameters based on a validation set.
→ **X**

P5 Self-SL: They do not consider using models pre-trained with Self-SL.
→ **X**

P5 Semi-SL: They do not consider using models trained with semi-supervised training paradigms.
→ **X**

Beck et al. (2021) Evaluate several AL methods in different settings to gain an understanding which AL methods outperform random queries. Further, they provide the Al toolkit DISTIL.

P1 Data Distribution: Perform experiments on CIFAR-10, CIFAR-100, Fashion-MNIST, SVHN and MNIST. → **X** due to no class imbalance.

P2 Starting Budget: They perform one experiment, where they evaluate a lower starting budget for MNIST.
→ (✔) due limited dataset.

P3 Query Size: They evaluate three different query sizes on CIFAR-10, but do so only for Random, Entropy and BADGE.
→ (✔) due to limited scope.

P4 Performant Baselines: Their supervised random baseline models perform good but no par with our random baseline models on CIFAR-10/100 (see section A.7).
→ (✔) due to limited scope.

P4 HP Optim. & Val. Split: They do not state optimizing their hyperparameters based on a validation set.
→ **X**

P5 Self-SL: They do not consider using models pre-trained with Self-SL.
→ **X**

P5 Semi-SL: They do not consider using models trained with semi-supervised training paradigms.

→ **X**

Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B Chan (2022a) Evaluate a multitude of different AL methods and provide the AL toolkit *DeepAL*$^+$.

P1 Data Distribution: Perform experiments on Tiny ImageNet, CIFAR-10 (and CIFAR-10 imbalanced), CIFAR-100, Fashion-MNIST, EMNIST and SVHN. Further Experiments are performed on an Histopathological image Classification Task (BreakHis) and Chest X-Ray Pneumonia classification (Pneumonia-MNIST) as well as the Waterbird dataset adopted from object recognition with correlated backgrounds.
→ ✔

P2 Starting Budget: They do not evaluate the performance under different starting budgets.
→ **X**

P3 Query Size: They evaluate multiple different query sizes on CIFAR-10 and analyze the difference.
→ ✔

P4 Performant Baselines: Their supervised random baseline models are not close to the performance of our random baseline models on CIFAR-10/100.
→ **X**

P4 HP Optim. & Val. Split: They do not state optimizing their hyperparameters based on a validation set.
→ **X**

P5 Self-SL: They do not consider using models pre-trained with Self-SL.
→ **X**

P5 Semi-SL: They do not consider using models trained with semi-supervised training paradigms.
→ **X**

Y.-C. Chan et al. (2021) Evaluate how AL methods interact with self- and semi-supervised training paradigms and how and whether they yield a benefit. The experiments in this paper differ from standard AL experiments by using only one query cycle, making it hard to compare systematically.

P1 Data Distribution: Perform experiments on CIFAR-10 and CIFAR-100.
→ **X**

P2 Starting Budget: Experiments are performed with a fixed starting budget of 3 samples per class or 2 samples per class in one case. This does not allow for evaluate of the influence of the starting budget on AL methods.
→ **X**

P3 Query Size: They query all samples for their final performance in one query step. This does not allow for evaluation of the influence of the query size on AL methods.
→ **X**

P4 Performant Baselines: Their supervised random baseline models perform on par with our random baseline models on CIFAR-10/100.
→ ✔

P4 HP Optim. & Val. Split: They do not state optimizing their hyperparameters based on a validation set.
→ **X**

P5 Self-SL: They use Debiased Contrastive Learning as self-supervised pretext task.
→ ✔

P5 Semi-SL: They use Pseudo-labeling and FixMatch as semi-supervised training paradigms.
→ ✔

A.2 Dataset details

Each dataset is split into a training, a validation and a test split.
For CIFAR-10/100 (LT) datasets the test split of size 10000 observations is already given and for MIO-TCD and ISIC-2019 we use a custom test split of 25% random observations of the entire dataset size. For MIO-TCD and ISIC-2019 the train, validation and test splits are imbalanced.
The validation split for all CIFAR-10 and CIFAR-100 datasets are 5000 randomly drawn observations corresponding to 10% of the entire dataset. For CIFAR-10 LT the validation split also consists of 5000 samples obtained from the dataset before the long-tail distribution is applied onto the training split. The CIFAR-10 LT validation split is therefore balanced. For MIO-TCD and ISIC-2019 the validation splits consist of 15% of the entire dataset.
The shared training & pool dataset for CIFAR-10/100 consists of 45000 observations. For CIFAR-10 LT the training & pool datasets consist of 12,600 observations. For MIO-TCD and ISIC-2019 the training & pool datasets consist of 60% the dataset.

A.2.1 Dataset descriptions

1. CIFAR-10: natural images containing 10 classes, label distribution is uniform
   Splits: (Train:45000; Val: 5000; Test; 10000)
   Whole Dataset: 60000

2. CIFAR-100: natural images containing 100 classes, label distribution is uniform
   Splits: (Train:45000; Val: 5000; Test; 10000)
   Whole Dataset: 60000

3. CIFAR-10 LT: natural images containing 10 classes, label distribution of test and validation split is uniform, label distribution of train split is artifically altered with imbalance factor $\rho = 50$ according to Cao et al., 2019. The resulting label distribution is shown in table A.1.
   Splits: (Train:$\sim$12,600; Val: 5000; Test; 10000)
   Whole Dataset: 27600

4. ISIC-2019: dermoscopic images containing 8 classes, label distribution of the dataset is imbalanced and shown in table A.2
   Splits: (Train:15200; Val: 3799; Test; 6332)
   Whole Dataset: 25331

5. MIO-TCD: natural images of traffic participants containing 11 classes, label distribution of the dataset is imbalanced and shown in table A.3
   Splits: (Train:311498; Val: 77875; Test; 129791)
   Whole Dataset: 519164

Table A.1: Number of Samples for each class in CIFAR-10 LT dataset. Validation and test sets are balanced.

Class	Train Split
airplane	4500
automobile (but not truck or pickup truck)	2913
bird	1886
cat	1221
deer	790
dog	512
frog	331
horse	214
ship	139
truck (but no pickup truck)	90

Table A.2: Number of Samples for each class in ISIC-2019

Class	Whole Dataset
Melanoma	4522
Melanocytic nevus	12875
Basal cell carcinoma	3323
Benign keratosis	867
Dermatofibroma	197
Vascular lesion	63
Squamos cell carcinoma	64

Table A.3: Number of samples for each class in MIO-TCD

Class	Whole Dataset
Articulated Truck	10346
Background	16000
Bicycle	2284
Bus	10316
Car	260518
Motorcycle	1982
Non-motorized vehicle	1751
Pedestrian	6262
Pickup truck	50906
Single unit truck	5120
Work van	9679

Table A.4: The exact values for all Label Regimes. Final Budget denotes the amount of labeled training samples at the end of the AL pipeline.

Dataset	CIFAR-10			CIFAR-100			CIFAR-10 LT			MIO-TCD			ISIC-2019		
Label Regime	Low	Medium	High	Low	Medium	High	Low	Medium	High	Low	Medium	High	Low	Medium	High
Starting Budget	50	250	1000	500	1000	5000	50	250	1000	55	275	1100	40	200	800
Query Size	50	250	1000	500	1000	5000	50	250	1000	55	275	1100	40	200	800
Final Budget	500	2500	10000	5000	10000	25000	500	2500	10000	550	2750	11000	400	2000	8000
Validation Set Size	250	1250	5000	2500	5000	5000	250	1250	5000	275	1375	5500	200	800	3799

A.3 Experimental setup, in more detail

Here we detail the most crucial information for reprocubility, re-implementation and checking our implementation. When in doubt, trust the information documented here with regard to what we wanted to do in our code.

A.3.1 Initial dataset setup

Before we do anything else the datasets are split according to fig. A.1 resulting in a tain split, a validation split and a test split. Each dataset has 3 different validation splits while always using the same test split. This is to ensure comparability across these splits without relying on cross-validation. The exact splits for each dataset are detailed in section A.2. After that the final datasets use for training and validation are then labeled according to the 'label strategy', which is described in fig. A.2. For all balanced datasets, we use class balanced label strategies since the label strategy only leads to different outcomes for imbalanced datasets. For CIFAR-10 LT we use the label strategy on the train split only, whereas for MIO-TCD and ISIC-2019 we use the label strategy on both train and validation split. The amount of data which is labeled for the final datasets of each split is then dependent upon the label-regime (described in more detail in section A.3.2).

A.3.2 Label Regimes

The exact Label Regimes are obtained by first taking the corresponding splits and then using the proper label strategy (see fig. A.2) in combination with the starting budget and validation set size according to table A.4.

A.3.3 Model architecture and training

On each training step the model is trained from its initialization to avoid a 'mode collapse' Kirsch, van Amersfoort, et al., 2019a. Further we select the checkpoint with the best validation set performance in the spirit of Gal, Islam, et al., 2017a. A ResNet-18 K. He, X. Zhang, et al., 2016 is the backbone for all of our experiments with weight decay disabled on bias parameters. If not otherwise noted, a nesterov momentum optimizer with momentum of 0.9 is used. For Self-SL models we use a two layer MLP as a classification head to make better use of the Self-SL representations with further details in section A.3.5. To obtain bayesian models we add dropout on the final representations before the classification head with probability ($p = 0.5$) following Gal, Islam, et al., 2017a. For all experiments on imbalanced datasets, we use the weighted CE-Loss following Munjal et al., 2022a based on the implementation in SK-Learn Buitinck et al., 2013 if not otherwise noted. Models trained purely on the labeled dataset upsample it to a size of 5500 following Kirsch, van Amersfoort, et al., 2019a if the labeled train set is smaller.

Bayesian Models All steps that require bayesian properties of the models including the prediction are obtained by drawing 50 MC samples following Gal, Islam, et al., 2017a; Kirsch, van Amersfoort, et al., 2019a.

ST ST models are trained for 200 epochs with Cosine Annealing and 10 epochs warmup.

Table A.5: HPs of the SimCLR pre-text training on each dataset. HP for CIFAR datasets are directly taken T. Chen et al., 2020 whereas MIO-TCD and ISIC-2019 HP are adapted from ImageNet experiments.

Dataset	CIFAR-10/CIFAR100/CIFAR-10 LT	MIO-TCD	ISIC-2019
Epochs	1000	200	1000
Optimizer	LARS	LARS	
Scheduler	Cosine Annealing	Cosine Annealing	
Warmup Epochs	10	10	
Temperature	0.5	0.1	
Batch Size	512	256	
Learning Rate	1	0.3	
Weight Decay	1E-4	1E-6	
Transform. Gauss Blur	False	True	
Transform. Color Jitter	Strength=0.5	Strength=1.0	

Self-SL Self-SL pre-trained models are trained for 80 epochs with a reduction of the learning rate with a factor of 10 every 20 epochs (MultiStepLR) and using a Mulit-Layer-Percpeptron (MLP) classification head (detailed description in section A.3.5). The complete setup of the training for SimCLR is described in section A.3.4.

Semi-SL Semi-SL training is identical to the one proposed with the FixMatch method Sohn, Berthelot, Carlini, et al., 2020, except that we do not use exponentially moving average models and restrict the training step from 1e6 to 2e5. The FixMatch implementation in our experiments is based on the open-source implementation of [2] and MixMatch for distribution alignment [3]. We always select the final Semi-SL model of the training for testing and querying. On imbalanced datasets we change the supervised term to the weighted CE-Loss and use distribution alignment on every dataset except for CIFAR-10 (where it does not improve performance Sohn, Berthelot, Carlini, et al., 2020). The HP sweep for our Semi-SL models includes weight decay and learning rate.

Hyperparameters All information with regard to the final HPs and our proposed methodology of finding them is detailed in section A.4

A.3.4 Self-supervised SimCLR pre-text training

Our implementation wraps the Pytorch-Lightning-Bolts implementation of SimCLR: https://lightning-bolts.readthedocs.io/en/latest/models/self_supervised.html#simclr . The training of our SimCLR models is performed by excluding the validation splits. Therefore three models are trained on each dataset, one for each different validation split. In table A.5 we give a list of the HPs used on each of our five different datasets. All other HPs are taken from T. Chen et al., 2020. Further, we did not optimize the HPs for SimCLR at all, meaning that on MIO-TCD and ISIC-2019 Self-SL models could perform even better than reported here.

A.3.5 MLP head for self-supervised pretrained models

The MLP Head used for the Self-SL models has 1 hidden layer of size 512 uses ReLU nonlinearities and BatchNorm. The results on CIFAR-10 based on which this design decision is based on is shown in table A.6.

[2] https://github.com/kekmodel/FixMatch-pytorch
[3] https://github.com/google-research/mixmatch

Table A.6: MLP Head Ablation for Self-SL models on CIFAR-10, over all labeled training set a small improvement for Multi-Layer-Perceptron is measurable compared to Linear classification head models. Reported as mean (std).

Labeled Train Set	Classification Head	Accuracy (Val) %	Accuracy (Test) %
50	Linear	69.87(1.62)	69.90(2.18)
50	2 Layer MLP	71.47(3.06)	71.54(0.56)
500	Linear	84.67(0.36)	83.51(0.45)
500	2 Layer MLP	85.37(0.16)	84.60(0.37)
1000	Linear	87.13(0.69)	85.97(0.64)
1000	2 Layer MLP	87.69(0.55)	86.57(0.42)
5000	Linear	90.77(0.44)	90.20(0.21)
5000	2 Layer MLP	91.12(0.32)	90.25(0.24)

A.3.6 List of data transformations

Standard The standard augmentations we use are based on the different datasets.
For CIFAR datasets these are in order of execution: RandomHorizontalFlip, RandomCrop to 32x32 with padding of size 4.

For MIO-TCD we use the standard ImageNet transformations: RandomResizedCrop to 224x224, Random Horizontal Flip.

For ISIC-2019 we use ISIC transformations which are: Resize to 300x300, RandomHorizontalFlip, RandomVerticalFlip, ColorJitter(0.02, 0.02, 0.02, 0.01) ,RandomRotation(rotation=(-180, 180), translate=(0.1, 0.1), scale=(0.7, 1.3)), RandomAffine(-180, 180), RandomCrop to 224x224.
These are based on the ISIC-2018 challenge best single model submission:
https://github.com/JiaxinZhuang/Skin-Lesion-Recognition.Pytorch

RandAugmentMC We use the same set of image transformations used in RandAugment Cubuk et al., 2020 with the parameters N=2 and M=10. A detailed list of image transformations alongside the corresponding values can be seen in Sohn, Berthelot, Carlini, et al., 2020 (Table 12).

The RandAugmentMC transformations were used additionally after the corresponding standard transformations for each dataset. RandAugmentMC(CIFAR) also adds cutout as a final transformation.

RandAugmentMC weak Works identical as RandAugmentMC and uses the same set of image transformations as for RandAugmentMC but changed its parameters to N=1 and M=2. Therefore the maximal range of values is divided by a factor of 5.

RandAugmentMC weak does not use cutout in difference to RandAugmentMC on CIFAR datasets.

A.3.7 Performance measure

As a measure of performance on CIFAR-10, CIFAR-100 and CIFAR-10 LT we use the accuracy while on MIO-TCD and ISIC-2019 we use balanced accuracy which is identical to mean recall shown in eq. (2.5).

$$\text{Mean Recall} = \sum_{c=1}^{C} \frac{1}{C} \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c} \tag{A.1}$$

Where C denotes the number of classes TP_c is the number of true positives for class c and FN_c being the number of samples belonging to class c being wrongly misclassified as another class.

A.3.8 Computational effort

Experiments were executed on a Cluster with access to multiple NVIDIA graphics cards. All ST and Self-SL pre-trained experiments used a single Nvidia RTX 2080 (10.7GB video-ram) graphic cards except for the BADGE experiments on CIFAR-100 and MIO-TCD which required more video-ram using Nvidia Titan RTX (23.6GB video-ram). The Semi-SL models on the CIFAR-10/100 (LT) datasets used also a single Nvidia RTX 2080 while on MIO-TCD and ISIC-2019 the Nvidia Titan RTX was utilized. For the results in our main table (excluding the HP optimization), the overall runtime was:

- All ST experiments: 1800 GPU hours
- All Self-SL pre-trained experiments: 1350 GPU hours[4]
- All Semi-SL experiments[5]: 11200 GPU hours

[4]Excluding the pre-training
[5]For only 2 Label Regimes and exluding MIO-TCD and ISIC-2019

A.4 Proposed hyperparameter optimization

Our proposed HP optimization for AL is based on the notion of minimizing HP selection effort by simplifying and reducing the search space. We use SGD Optimizer with Nesterov momentum of 0.9 and select a number of epochs that always allow a complete fit of the model. The scheduler is also fixed across experiments as are the warmup epochs if used. Secondly, we pre-select the batchsize for each dataset since it is usually not a critical HP as long as it is big enough for BatchNorm to work properly.

ST For our ST models the final HP for each dataset and Label Regime are shown in table A.7.
HP sweep: weight decay: (5E-3, 5E-4); learning rate: (0.1, 0.01); data transformation: (RandAugmentMC, Standard)

Self-SL For our Self-SL pre-trained models the final HP for each dataset and Label Regime are shown in table A.8.
HP sweep: weight decay: (5E-3, 5E-4); learning rate: (0.01, 0.001); data transformation: (RandAugmentMC weak, Standard)

Semi-SL For our Semi-SL models we follow Sohn, Berthelot, Carlini, et al., 2020 with regard to HP selection as closely as possible. The final HP for each dataset and Label Regime are shown in table A.9.
HP sweep: weight decay and learning rate.

Table A.7: Final HPs for each dataset and Label Regime for our ST models based on our HP tuning. HPs denoted with a * are fixed across datasets and HP denoted with a + are pre-selected for each dataset while all other HP are obtained via sweeping.

Dataset	CIFAR-10			CIFAR-100			CIFAR-10 LT			MIO-TCD			ISIC-2019		
Label Regime	Low	Medium	High	Low	Medium	High	Low	Medium	High	Low	Medium	High	Low	Medium	High
Epochs*		200			200			200			200			200	
Optimizer*		SGD Nesterov 0.9			SGD Nesterov 0.9			SGD Nesterov 0.9			SGD Nesterov 0.9			SGD Nesterov 0.9	
Scheduler*		Cosine Annealing			Cosine Annealing			Cosine Annealing			Cosine Annealing			Cosine Annealing	
Warmup Epochs*		10			10			10			10			10	
Loss*		CE-Loss			CE-Loss			CE-Loss			CE-Loss			CE-Loss	
Sampling+		standard			standard			oversampling		oversampling			oversampling		
Batch Size+		1024			1024			1024			512			512	
Learning Rate		0.1			0.1			0.1		0.01	0.01	0.1		0.1	
Weight Decay		5E-3			5E-3			5E-3		5E-3	5E-4	5E-3		5E-3	
Data Augmentation		RandAugmentMC (CIFAR)			RandAugmentMC (CIFAR)			RandAugmentMC (CIFAR)			RandAugmentMC (ImageNet)			RandAugmentMC (ISIC)	

Table A.8: Final HPs for each dataset and Label Regime for our Self-SL models based on our HP tuning. Overall Performance was remarkably stable with regard to HPs and stronger augmentations did not necessarily improve performance in the same way as for ST models. This is presumably due to the pre-trained representations. HP denoted with a * are fixed across datasets and HP denoted with a + are pre-selected for each dataset while all other HP are obtained via sweeping.

Dataset	CIFAR-10			CIFAR-100			CIFAR-10 LT			MIO-TCD				ISIC-2019		
Label Regime	Low	Medium	High	Low	Medium	High	Low	Medium	High	Low	Medium	High		Low	Medium	High
Epochs*		80			80			80			80				80	
Optimizer*		SGD Nesterov 0.9			SGD Nesterov 0.9			SGD Nesterov 0.9			SGD Nesterov 0.9				SGD Nesterov 0.9	
Scheduler*		MulitStepLR			MulitStepLR			MulitStepLR			MulitStepLR				MulitStepLR	
Warmup Epochs*		0			0			0			0				0	
Loss*		CE-Loss			CE-Loss			CE-Loss			CE-Loss				CE-Loss	
Sampling+		standard			standard			oversampling		oversampling				oversampling		
Batch Size+		64			64			64			256				128	
Learning Rate		0.001			0.001		0.01	0.01	0.001	0.001	0.01	0.001		0.01	0.001	0.01
Weight Decay		5E-3			5E-3		5E-4	5E-4	5E-3		5E-3			5e-3	5E-4	5e-3
Data Augmentation		RandAugmentMC weak (CIFAR)			RandAugmentMC weak (CIFAR)			Standard (CIFAR)			RandAugmentMC weak (ImageNet)	Standard (ImageNet)	Standard (ImageNet)		Standard (ISIC)	

Table A.9: Final HPs for each dataset and Label Regime for our Semi-SL models based on our HP tuning. HP denoted with a * are fixed across datasets and HP denoted with a + are pre-selected for each dataset while all other HP are obtained via sweeping. – denotes not performed experiments.

Dataset	CIFAR-10			CIFAR-100			CIFAR-10 LT			MIO-TCD			ISIC-2019		
Label Regime	Low	Medium	High	Low	Medium	High	Low	Medium	High	Low	Medium	High	Low	Medium	High
Optimization Steps*	2E5		–	2E5		–	2E5		–	2E5		–	2E5		–
Optimizer*	SGD Nesterov 0.9		–	SGD Nesterov 0.9		–	SGD Nesterov 0.9		–	SGD Nesterov 0.9		–	SGD Nesterov 0.9		–
Scheduler*	Cosine Annealing		–	Cosine Annealing		–	Cosine Annealing		–	Cosine Annealing		–	Cosine Annealing		–
Warmup Steps+	0		–	0		–	0		–	3000		–	3000		–
Loss+	CE-Loss		–	CE-Loss		–	weigthed CE-Loss		–	weigthed CE-Loss		–	weigthed CE-Loss		–
Sampling*	standard		–	standard		–	standard		–	standard		–	standard		–
λ_u^*	1		–	1		–	1		–	1		–	1		–
μ^*	7		–	7		–	7		–	7		–	7		–
τ^*	0.95		–	0.95		–	0.95		–	0.95		–	0.95		–
Distribution Alignment+	False		–	True		–	True		–	True		–	True		–
Batch Size*	64		–	64		–	64		–	64		–	64		–
Learning Rate	0.03		–	0.03		–	0.03		–			–			–
Weight Decay	5E-4		–	5E-4		–	1E-3	5E-4	–			–			–
Data Augmentation+	Standard (CIFAR)		–	Standard (CIFAR)		–	Standard (CIFAR)		–	Standard (ImageNet)		–	Standard (ISIC)		–
Unlabeled Augmentation+	RandAugmentMC (CIFAR)		–	RandAugmentMC (CIFAR)		–	RandAugmentMC (CIFAR)		–	RandAugmentMC (ImageNet)		–	RandAugmentMC (ISIC)		–

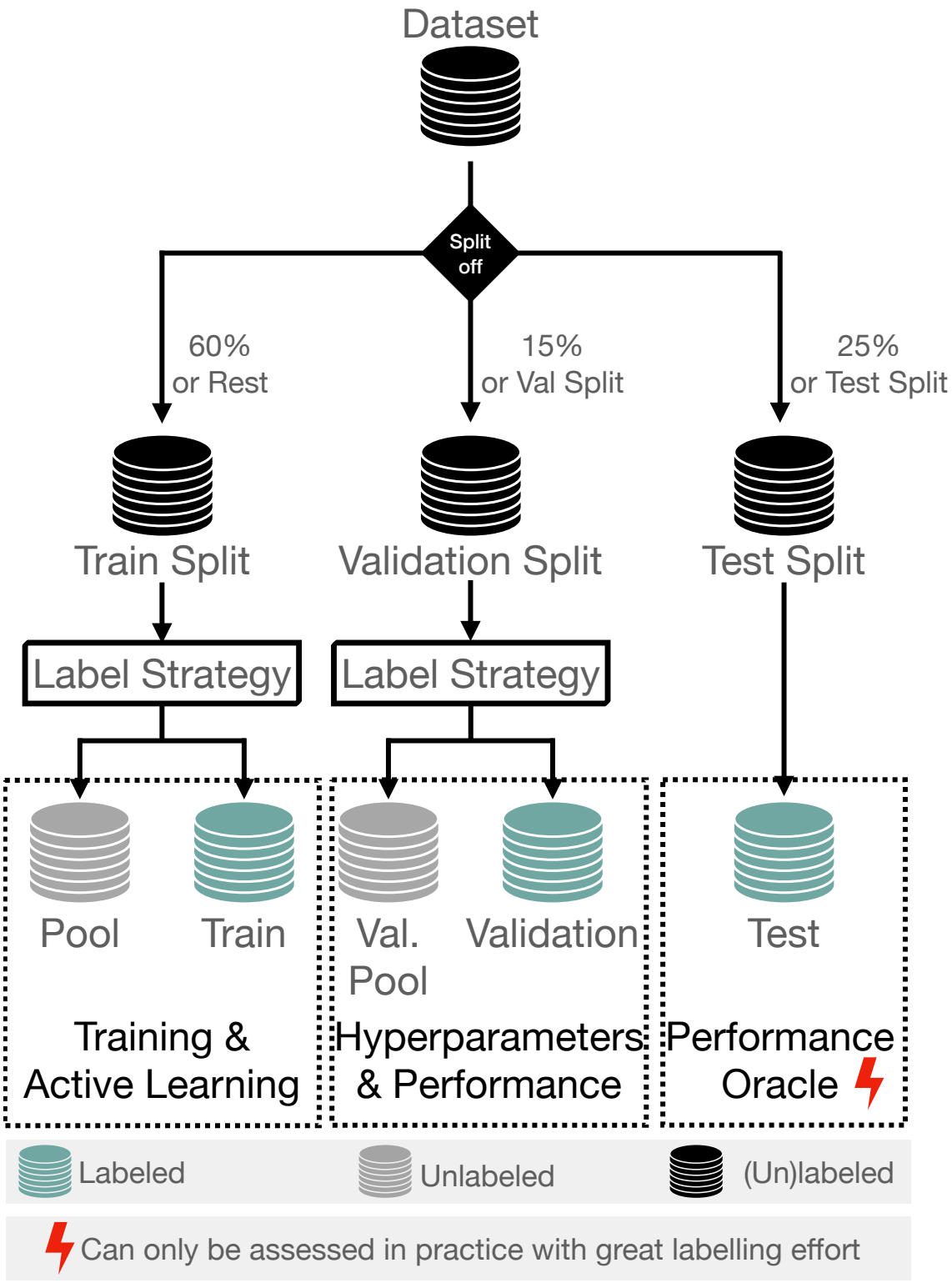


Figure A.1: Description of the three different data splits and their use-cases. The complete separation of a validation split allows to compare across Label Regimes and incorporate techniques for performance evaluation s.a. Active Testing Kossen et al., 2021. For evaluation and development the test split should be as big as possible since QM recommendations are based on the test set performance making it a form of "oracle". An estimate of the size a dataset is required to have to measure specific performance differences can be derived using Hoeffding's inequality Hoeffding, 1994; Oliver et al., 2019.

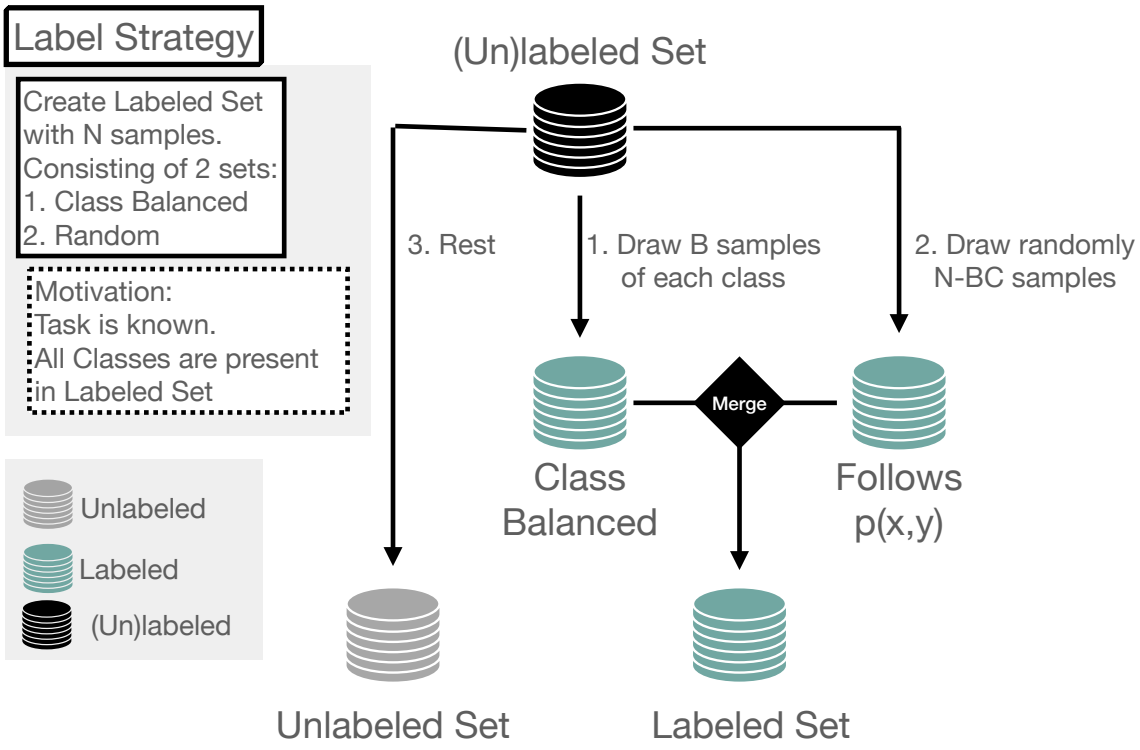


Figure A.2: The Label Strategy used on the two roll-out datasets MIO-TCD and ISIC-2019 and for train and pool set on CIFAR-10 LT. For class balanced datasets this strategy does not induce meaningful changes to balanced starting budgets.

A.5 Detailed results

A.5.1 Main results

General observations: For all datasets, the overall performance of models was primarily determined by the training strategy and the HP selection, with the benefits of AL being generally smaller compared to the proper selection of both. For the three toy datasets, Semi-SL generally performed best, followed by Self-SL and ST last, whereas, for the two real-world datasets, Semi-SL showed no substantial improvement over ST in the first training stage and, therefore, further runs were omitted. Also, the absolute performance gains for Self-SL models with AL are generally smaller compared to ST models. For Semi-SL, there were generally only very small performance gains or substantially worse performance with AL observed. Concerning the effect of AL, the high-Label Regime proved to work for ST models on all datasets and Self-SL models. On the two real-world datasets, MIO-TCD and ISIC-2019, a dip in performance at 7k samples for all ST models could be observed. This behavior is ablated in section A.8.1.

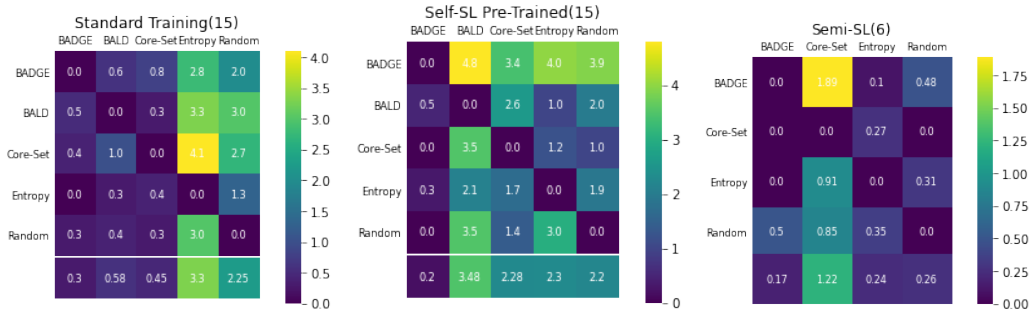
Evaluation using the pair-wise penalty matrix: We use the pair-wise penalty matrix (PPM) to compare whether the performance of one query method significantly outperforms the others. It is essentially a measure of how often one method significantly outperforms another method based on a t-test with $\alpha = 0.05$ (more info in J. T. Ash et al., 2020; Beck et al., 2021). This allows to aggregate results over different datasets and Label Regimes, with the disadvantage being that the absolute performance is not taken into consideration. When reading a PPM, each row i indicates the number of settings in which method i beats other methods, while column j indicates the number of settings in which method j is beaten by another method.

We show the PPMs aggregated over all datasets and Label Regimes for each training paradigm in fig. A.3.

For all methods, BADGE is the QM that is least often outperformed by other QMs. Further, for Self-SL models, it is never significantly outperformed by Random, whereas it is seldomly significantly outperformed for ST models. Based on this, we deem BADGE to be the best of our compared QMs for both ST and Self-SL models. Since BADGE is more often outperformed by Random (0.5) on the Semi-SL datasets and the additional high training cost for each iteration, we believe that Random is the better choice in many cases.

Evaluation using the area under the budget curve: For each of the following subsections, we added the area under the budget curve (AUBC) for each dataset and Label Regime to allow assessing the absolute performance each QM brings. Generally, higher values are better. For more information, we refer to Zhan, H. Liu, et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B Chan, 2022a.

The results on the dataset for AUBC also show that BADGE is always one of the best performing AL methods. This is in line with the findings based on the PPM.



(a) (b) (c)

Figure A.3: PPMs aggregated over all experiments for Standard Models (a), Self-Sl Pre-Trained Models (b) and Semi-SL models(c).
The value in the title (X) gives the highest possible value in a cell and the lowest row is the mean value across a column j without row $i = j$ signaling how often on average on QM is outperformed by another.

CIFAR-10

The AUBC values are shown in table A.10.

Table A.10: Area Under Budget Curve values for CIFAR-10.

	Label Regime	Low-Label		Medium-Label		High-Label	
		Mean	STD	Mean	STD	Mean	STD
Training	Query Method						
ST	BADGE	0.4730	0.0105	0.7289	0.0025	0.8599	0.0016
	BALD	0.4744	0.0106	0.7253	0.0051	0.8578	0.0017
	Entropy	0.4307	0.0018	0.6859	0.0042	0.8498	0.0025
	Core-Set	0.4681	0.0038	0.7282	0.0043	0.8629	0.0017
	Random	0.4720	0.0144	0.7309	0.0068	0.8526	0.0030
Self-SL	BADGE	0.8282	0.0016	0.8728	0.0018	0.9086	0.0012
	BALD	0.8005	0.0056	0.8692	0.0011	0.9093	0.0010
	Entropy	0.8002	0.0090	0.8663	0.0017	0.9071	0.0010
	Core-Set	0.8224	0.0026	0.8670	0.0009	0.9015	0.0009
	Random	0.8117	0.0040	0.8669	0.0009	0.8989	0.0007
Semi-SL	BADGE	0.9349	0.0010	0.9488	0.0022	–	–
	Entropy	0.9193	0.0082	0.9497	0.0007	–	–
	Core-Set	0.9343	0.0018	0.9442	0.0007	–	–
	Random	0.9326	0.0050	0.9478	0.0002	–	–

CIFAR-100

The AUBC values are shown in table A.11.

Table A.11: Area Under Budget Curve values for CIFAR-100.

	Label Regime	Low-Label		Medium-Label		High-Label	
		Mean	STD	Mean	STD	Mean	STD
Training	Query Method						
ST	BADGE	0.3525	0.0042	0.4855	0.0044	0.6627	0.0017
	BALD	0.3586	0.0007	0.4865	0.0019	0.6658	0.0002
	Entropy	0.3036	0.0095	0.4440	0.0007	0.6569	0.0021
	Core-Set	0.3458	0.0010	0.4767	0.0031	0.6560	0.0004
	Random	0.3599	0.0027	0.4791	0.0044	0.6474	0.0015
Self-SL	BADGE	0.5397	0.0030	0.6020	0.0019	0.6858	0.0021
	BALD	0.5028	0.0043	0.5754	0.0027	0.6784	0.0009
	Entropy	0.5111	0.0066	0.5857	0.0028	0.6857	0.0032
	Core-Set	0.5337	0.0044	0.5917	0.0032	0.6804	0.0013
	Random	0.5365	0.0017	0.5970	0.0017	0.6757	0.0013
Semi-SL	BADGE	0.5562	0.0033	0.6222	0.0041	–	–
	Entropy	0.5328	0.0158	0.6152	0.0038	–	–
	Core-Set	0.5220	0.0101	0.6083	0.0061	–	–
	Random	0.5713	0.0066	0.6307	0.0016	–	–

CIFAR-10 LT

The AUBC values are shown in table A.12.

Table A.12: Area Under Budget Curve values for CIFAR-10 LT.

	Label Regime	Low-Label		Medium-Label		High-Label	
		Mean	STD	Mean	STD	Mean	STD
Training	Query Method						
ST	BADGE	0.3788	0.0152	0.5887	0.0098	0.7577	0.0004
	BALD	0.3919	0.0111	0.5935	0.0141	0.7590	0.0034
	Entropy	0.3740	0.0064	0.5793	0.0114	0.7454	0.0023
	Core-Set	0.3939	0.0249	0.6162	0.0200	0.7667	0.0058
	Random	0.3615	0.0118	0.5446	0.0214	0.7263	0.0044
Self-SL	BADGE	0.5373	0.0233	0.6501	0.0026	0.7704	0.0093
	BALD	0.5431	0.0202	0.6549	0.0097	0.7742	0.0036
	Entropy	0.5282	0.0225	0.6450	0.0090	0.7707	0.0061
	Core-Set	0.5298	0.0182	0.6171	0.0154	0.7555	0.0032
	Random	0.5397	0.0208	0.6173	0.0097	0.7554	0.0069
Semi-SL	BADGE	0.7233	0.0166	0.7616	0.0087	–	–
	Entropy	0.6934	0.0289	0.7590	0.0101	–	–
	Core-Set	0.6825	0.0103	0.7608	0.0108	–	–
	Random	0.6965	0.0264	0.7363	0.0077	–	–

MIO-TCD

The AUBC values are shown in table A.13.

Table A.13: Area Under Budget Curve values for MIO-TCD.

	Label Regime	Low-Label		Medium-Label		High-Label	
		Mean	STD	Mean	STD	Mean	STD
Training	Query Method						
ST	BADGE	0.3539	0.0041	0.4688	0.0153	0.6614	0.0080
	BALD	0.3254	0.0155	0.4830	0.0092	0.6884	0.0104
	Entropy	0.3201	0.0097	0.4134	0.0230	0.6078	0.0176
	Core-Set	0.3514	0.0134	0.4678	0.0181	0.7098	0.0056
	Random	0.3510	0.0151	0.4564	0.0140	0.6065	0.0120
Self-SL	BADGE	0.5446	0.0122	0.6365	0.0054	0.7174	0.0040
	BALD	0.4494	0.0102	0.5741	0.0138	0.6041	0.0092
	Entropy	0.5105	0.0092	0.6416	0.0075	0.6972	0.0029
	Core-Set	0.5060	0.0082	0.5900	0.0190	0.6699	0.0166
	Random	0.5298	0.0109	0.6124	0.0032	0.6975	0.0054

ISIC-2019

The AUBC values are shown in table A.14.

Table A.14: Area Under Budget Curve values for ISIC-2019.

	Label Regime	Low-Label		Medium-Label		High-Label	
		Mean	STD	Mean	STD	Mean	STD
Training	Query Method						
ST	BADGE	0.3204	0.0099	0.4331	0.0146	0.5628	0.0101
	BALD	0.3190	0.0133	0.4521	0.0211	0.5534	0.0052
	Entropy	0.3241	0.0067	0.4207	0.0335	0.5631	0.0061
	Core-Set	0.3426	0.0099	0.4501	0.0139	0.5708	0.0096
	Random	0.3376	0.0243	0.4116	0.0201	0.5273	0.0048
Self-SL	BADGE	0.3809	0.0168	0.4679	0.0174	0.5761	0.0063
	BALD	0.3949	0.0209	0.4847	0.0018	0.5914	0.0080
	Entropy	0.3666	0.0165	0.4659	0.0104	0.5872	0.0096
	Core-Set	0.3752	0.0205	0.4472	0.0071	0.5556	0.0069
	Random	0.3736	0.0092	0.4555	0.0053	0.5547	0.0066

A.5.2 Low-Label Query Size

To investigate the effect of query size in the low-Label Regime, we conduct an ablation with Self-SL pre-trained models on CIFAR-100 and ISIC-2019. For CIFAR-100 the query sizes are 50, 500 and 2000, while for ISIC-2019 they are 10, 40 and 160.

Here the accuracies for the overlapping labeled samples are shown which are analyzed using a t-test.

CIFAR100 The results are shown in table A.15 and table A.16. When comparing the performance of the same QM using different query sizes only BALD and Core-Set lead to statistically significant difference in performance. While it is consistent across both comparisons for BALD with the the performance difference widening for larger labeled sets and more iterations the trend for Core-Set is not as clear.

ISIC-2019 The results are shown in table A.17 and table A.18. Entropy is the only QM showing a significant difference, indicating that a query size of 40 outperforms a query size of 10 for a training set size of 160. However, this behavior does not extend to the other training set sizes. Whereas, for BALD, there is a consistent trend that smaller query sizes lead to increased performance.

Table A.15: Accuracies % for the low-label comparison for CIFAR100 with query sizes 50 and 500 at overlapping training set sizes. Reported as mean (std). Values with a significant difference (t-test) across query sizes are denoted with $*$.

Labeled Samples	1000		1500	
Query Size	50	500	50	500
BADGE	45.75 (0.75)	46.32 (0.20)	50.02 (0.92)	50.24 (0.61)
BALD	45.54 (0.20)	42.31 (1.71)	49.41 (0.39)*	46.00 (0.44)*
Core-Set	46.53 (0.76)	46.02 (0.74)	49.34 (0.72)	49.70 (0.60)
Entropy	43.82 (0.40)	42.95 (1.32)	47.05 (0.91)	46.75 (0.76)
Random	46.10 (0.41)	45.22 (0.34)	49.88 (0.26)	49.79 (0.27)

Table A.16: Accuracies % for the low-label comparison for CIFAR100 with query sizes 500 and 2000 at overlapping training set sizes. Reported as mean (std). Values with a significant difference (t-test) across query sizes are denoted with $*$.

Labeled Samples	2500		4500	
Query Size	500	2000	500	2000
BADGE	54.62 (0.60)	55.03 (0.38)	50.92 (0.12)	59.98 (0.60)
BALD	50.92 (0.30)	49.96 (1.43)	55.86 (0.21)*	52.14 (1.36)*
Core-Set	54.72 (0.16)*	53.52 (0.33)*	58.71 (0.61)	58.28 (0.43)
Entropy	51.41 (0.53)	51.79 (0.76)	57.12 (0.53)	57.53 (0.43)
Random	54.71 (0.38)	54.38 (0.21)	59.98 (0.60)	59.48 (0.48)

Table A.17: Accuracies % for the low-label comparison for ISIC-2019 with query sizes 10 and 40 at overlapping training set sizes. Reported as mean (std). Values with a significant difference (t-test) across query sizes are denoted with $*$.

Labeled Sample	80		120		160		200		240	
Query Size	10	40	10	40	10	40	10	40	10	40
BADGE	34.39 (0.86)	36.03 (0.39)	37.58 (1.93)	36.40 (1.10)	37.58 (2.51)	37.24 (2.15)	38.32 (1.59)	37.52 (1.16)	38.78 (1.64)	38.94 (2.97)
BALD	38.11 (1.42)	36.51 (1.26)	38.80 (1.19)	37.26 (4.90)	40.40 (1.76)	39.23(1.89)	42.15 (1.49)	40.09 (2.59)	42.26 (0.97)	40.18 (2.70)
Core-Set	35.57 (1.49)	35.38 (0.50)	36.87 (1.73)	36.52 (2.24)	38.55 (2.37)	38.19 (1.69)	38.41 (1.80)	37.19 (2.93)	39.04 (2.94)	38.30 (1.66)
Entropy	35.19 (0.63)	34.87 (0.91)	37.06 (1.63)	38.59 (2.06)	36.95 (1.13)*	39.67 (0.55)*	38.25 (3.11)	39.93 (2.38)	39.38 (2.62)	40.26 (2.49)
Random	36.10 (0.70)	35.69 (0.68)	37.57 (1.79)	37.65 (2.15)	37.43 (1.77)	40.62 (0.62)	38.41 (1.46)	40.29 (1.78)	3835 (2.43)	41.14 (0.37)

Table A.18: Accuracies % for the low-label comparison for ISIC-2019 with query sizes 40 and 160 at overlapping training set sizes. Reported as mean (std). Values with a significant difference (t-test) across query sizes are denoted with $*$.

Labeled Samples	200		360	
Query Size	40	160	40	160
BADGE	37.52 (1.16)	38.11 (1.07)	42.90 (3.29)	42.86 (1.44)
BALD	40.09 (2.59)	39.67 (2.54)	44.49 (1.78)	42.83 (1.05)
Core-Set	37.79 (2.93)	37.71 (3.18)	39.56 (1.48)	39.47 (1.04)
Entropy	39.93 (2.38)	37.80 (1.46)	42.03 (1.16)	40.82 (1.46)
Random	40.29 (1.78)	40.80 (2.34)	42.10 (2.29)	42.10 (2.29)

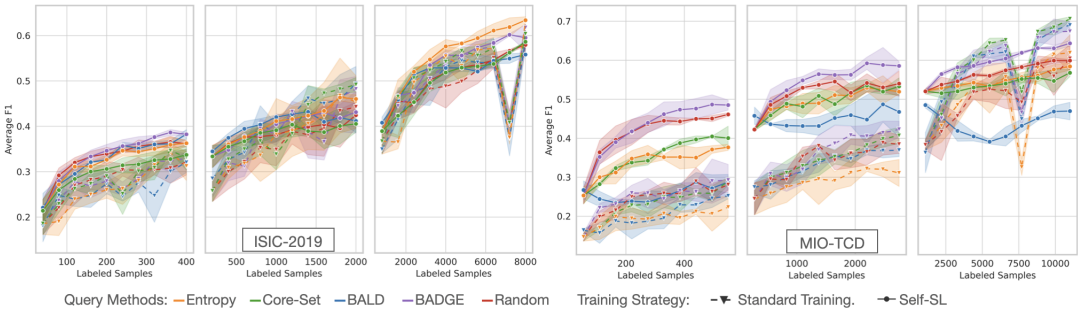


Figure A.4: Macro-averaged F1-scores for ISIC-2019 and MIO-TCD datasets. Across the board, BADGE is still the best-performing QM which can also be seen in Table A.19. Please interpret the results with care, as the model configurations are optimized for balanced accuracy.

Table A.19: Area Under Budget Curve values based on macro-averaged F1-scores for ISIC-2019 and MIO-TCD (corresponding plots in Figure A.4).

(a) ISIC-2019

	Label Regime	Low-Label		Medium-Label		High-Label	
		Mean	STD	Mean	STD	Mean	STD
Training	Query Method						
Standard Training	BADGE	0.2791	0.0081	0.3810	0.0167	0.5059	0.0054
	BALD	0.2587	0.0196	0.4050	0.0216	0.4866	0.0100
	Entropy	0.2652	0.0147	0.3763	0.0453	0.4988	0.0056
	Core-Set	0.2800	0.0105	0.3937	0.0082	0.5139	0.0141
	Random	0.2817	0.0153	0.3638	0.0292	0.4774	0.0035
Self-SL Pre-Trained	BADGE	0.3390	0.0107	0.3978	0.0053	0.5380	0.0067
	BALD	0.3273	0.0054	0.4065	0.0047	0.5173	0.0008
	Entropy	0.3232	0.0039	0.4081	0.0061	0.5586	0.0143
	Core-Set	0.3010	0.0129	0.3833	0.0094	0.5045	0.0123
	Random	0.3333	0.0033	0.3852	0.0153	0.5100	0.0059

(b) MIO-TCD

	Label Regime	Low-Label		Medium-Label		High-Label	
		Mean	STD	Mean	STD	Mean	STD
Training	Query Method						
Standard Training	BADGE	0.2503	0.0111	0.3604	0.0249	0.5759	0.0075
	BALD	0.2023	0.0127	0.3325	0.0082	0.5758	0.0116
	Entropy	0.1957	0.0208	0.2929	0.0218	0.5066	0.0290
	Core-Set	0.2301	0.0155	0.3450	0.0203	0.5945	0.0099
	Random	0.2446	0.0091	0.3438	0.0132	0.5085	0.0093
Self-SL Pre-Trained	BADGE	0.4295	0.0205	0.5487	0.0129	0.5994	0.0035
	BALD	0.2560	0.0132	0.4488	0.0236	0.4335	0.0088
	Entropy	0.3398	0.0104	0.4977	0.0095	0.5481	0.0078
	Core-Set	0.3525	0.0066	0.4970	0.0160	0.5375	0.0138
	Random	0.4175	0.0158	0.5194	0.0101	0.5681	0.0036

A.5.3 Macro averaged F1-scores

Additionally, we provide the macro-averaged F1-scores for ISIC-2019 and MIO-TCD dataset. A plot showing the performance of both ST and Self-SL models is shown in fig. A.4 and the resulting AUBC values are shown in table A.19.

A.5.4 Semi-Supervised Learning

Results of FixMatch for all HPs on the whole validation splits are shown separately for MIO-TCD in table A.20 and ISIC-2019 in table A.21. Based on the performance which did not improve substantially over even ST models we decided to omit all further AL experiments.

Table A.20: MIO-TCD FixMatch results reported on the test sets (balanced accuracy in %). Reported as mean (std).

FixMatch Sweep MIO-TCD Labeled Train Samples	Learning Rate	Weight Decay	Balanced Accuracy (Test)
55	0.3	5E-3	18.1(1.1)
		5E-4	28.0(0.4)
	0.03	5E-3	26.8(0.5)
		5E-4	29.7(2.4)
275	0.3	5E-3	20.2(6.2)
		5E-4	28.2(1.5)
	0.03	5E-3	31.4(0.7)
		5E-4	36.0(1.8)

Table A.21: ISIC-2019 FixMatch results reported on the test sets (balanced accuracy in %). Reported as mean (std).

FixMatch Sweep ISIC-2019			
Labeled Train Samples	Learning Rate	Weight Decay	Balanced Accuracy (Test)
40	0.3	5E-3	14.0(2.5)
		5E-4	26.9(1.5)
	0.03	5E-3	24.5(4.4)
		5E-4	31.3(1.7)
200	0.3	5E-3	15.2(3.6)
		5E-4	19.1(1.5)
	0.03	5E-3	26.3(0.8)
		5E-4	24.3(3.6)

A.6 Discussion and further observations

The results are interpreted based on the assumption that a QM performing on a similar level as Random is not a drawback as long as it brings in other settings performance improvements over random queries. This mostly follows in line with the PPM as a performance metric but mostly focuses on the row that compares each QM with random queries. However, if a QM shows behavior leading to much worse behavior than random as Entropy does or shows signs of the cold start problem, we deem this as highly problematic. In these settings, one loses significant performance whilst paying a cost in computing and setup corresponding to AL. Therefore, we use random queries as a baseline for all QMs.
Based on this our recommendation for BADGE is given for Self-SL and ST trainings.
The main disadvantage of this approach is that absolute performance difference are not captured in this aggregated format.

A.7 Comparing random-sampling baselines across studies

Here we compare the performance of random-sampling baselines on the most commonly utilized dataset CIFAR-10 and CIFAR-100 across different studies for ST, Self-SL and Semi-SL models along strategic point where overlap in between papers occurs. For CIFAR-10 the results of this comparison are shown for the high-Label Regime in table A.22 and the low- and mid-Label Regime in table A.23. Similarly for CIFAR-100 the results are shown in table A.24 for the high-Label Regime and table A.25 for the low- and mid-Label Regimes. Overall our ST random baselines outperform all other random baselines. Our Self-SL models also outperform the only other relevant literature Bengar et al., 2021 on CIFAR-10. Further, our Semi-SL models also outperform the relevant literature Mittal, Tatarchenko, et al., 2019a; Gao, Z. Zhang, Yu, Arık, et al., 2020 on CIFAR-10 and CIFAR-100.

Table A.22: Comparison of random baseline model accuracy in % on the test set for the high label-regime for CIFAR-10 across different papers. Best performing models for each training strategy are **highlighted**. Values denoted with – represent not performed experiments. Values with a denoted with a * are reprinted from Munjal et al., 2022a. Values which are sourced from a graph are subject to human read-out error.

Information				Number Labeled Training Samples					
Paper	Training	Model	Source	1k	2k	5k	10k	15k	20k
QBC	ST	DenseNet121	Graph			74*	82.5*	-	-
VAAL	ST	VGG16	Graph	-	-	61.35*	68.17*	72.96*	75.99*
CoreSet	ST	VGG16	Graph	-	-	60*	68*	71*	74*
Agarwal et al.	ST	VGG16	Graph	-	-	61.5	68	72	76
Munjal-SR	ST	VGG16	Table	-	-	82.16	85.07	89.43	91.16
Mittal et al.	ST	WRN28-2	Graph	57	73	82.5	86	90.7	92
LLAL	ST	ResNet18	Graph	51	63	81*	87*	-	-
CoreCGN	ST	ResNet18	Graph	50	64	80*	85.5*	-	-
TA-VAAL	ST	ResNet18	Graph	50	65	81*	87.5*	-	-
Krishnan et al.	ST	ResNet18	Graph	47	60	78	86	-	-
Yi et al.	ST	ResNet18	Graph	47.5	56	78	86	-	-
Bengar et al.	ST	ResNet18	Graph	45	55	73	81	85	88
Beck et al.	ST	ResNet18	Graph	55	-	-	84	85	90.5
Zhan et al.	ST	ResNet18	Graph	45	-	-	76	-	-
Munjal-SR	ST	ResNet18	Table	-	-	84.69	88.45	89.98	92.29
Ours	ST	ResNet18	Table	**72.4**	**79.8**	**85.5**	**90.5**	-	-
Bengar et al.	Self-SL	ResNet18	Graph	**87**	88	89.5	90.5.	91	91.5
Ours	Self-SL	ResNet18	Table	86.2	**88.3**	**90.1**	**91.4**	-	-
Mittal et al.	Semi-SL	WRN28-2	Graph	88	91	92.5	93.8	94	94.5
Gao et al.	Semi-SL	WRN28-2	Graph	91.5	91	-	-	-	-
Ours	Semi-SL	ResNet18	Table	**94.7**	**95.0**	-	-	-	-

Table A.23: Comparison of random baseline model accuracy in % on the test set for the low- and mid-Label Regime for CIFAR-10 across different papers. Best performing models for each training strategy are **highlighted**. Values denoted with – represent not performed experiments. Values which are sourced from a graph are subject to human read-out error.

Information				Number Labeled Training Samples				
Paper	Training	Model	Source	50	100	200	250	500
Chan et al.	ST	WRN28-2	Table	-	-	-	40.9	-
Mittal et al.	ST	WRN28-2	Graph	-	-	-	36	48
Bengar et al.	ST	ResNet18	Graph	-	-	-	-	38
Ours	ST	ResNet18	Table	**25.1**	**32.3**	**44.4**	**47.0**	**61.2**
Chan et al.	Self-SL	WRN28-2	Table	-	-	-	76.7	-
Bengar et al.	Self-SL	ResNet18	Graph	62	**77**	81	83	85
Ours	Self-SL	ResNet18	Table	**71.3**	76.8	**81.2**	81.4	84.1
Chan et al.	Semi-SL	WRN28-2	Table	-	-	-	**93.1**	-
Mittal et al.	Semi-SL	WRN28-2	Graph	-	-	-	82	85
Gao et al.	Semi-SL	WRN28-2	Table	-	47.9	89.2	90.2	-
Ours	Semi-SL	ResNet18	Graph	**90**	**91**	**93**	**93**	**94**

Table A.24: Comparison of random baseline model accuracy in % on the test set for the high-Label Regime for CIFAR-100 across different papers. Best performing models for each training strategy are **highlighted**. Values denoted with – represent not performed experiments. Values which are sourced from a graph are subject to human read-out error.

Information				Number Labeled Training Samples			
Paper	Training	Model	Source	5k	10k	15k	20k
Agarwal et al.	ST	VGG16	Graph	28	35	41.5	46
Agarwal et al.	ST	ResNet18	Graph	29.5	38	45	49
Core-Set	ST	VGG16	Graph	27	37	42	49
VAAL	ST	VGG16	Graph	28	35	42	46
Munjal et al.	ST	VGG16	Graph	39.44	49	55	59
VAAL	ST	ResNet18	Graph	28	38	45	49
TA-VAAL	ST	ResNet18	Graph	43	52	60	63.5
Bengar et al.	ST	ResNet18	Graph	27	45	52	58
Beck et al.	ST	ResNet18	Graph	40	53	60	64
Zhan et al.	ST	ResNet18	Graph	-	39	-	-
Munjal et al.	ST	ResNet18	Table	?	61.1	**66.9**	69.8
Mittal et al.	ST	WRN28-2	Graph	44.9	58	64	68
Ours	ST	ResNet18	Table	**49.2**	**61.3**	66.7	**70.2**
Bengar et al.	Self-SL	ResNet18	Table	60	63	63.5	64
Ours	Self-SL	ResNet18	Table	**60.4**	**64.8**	**68.4**	**70.7**
Mittal et al.	Semi-SL	WRN28-2	Graph	59	65	70	71
Gao et al.	Semi-SL	WRN28-2	Table	**63.4**	67	68	70
Ours	Semi-SL	ResNet18	Graph	**63.5**	**68.5**	-	-

Table A.25: Comparison of random baseline model accuracy in % on the test set for the low- and mid- Label Regime for CIFAR-100 across different papers. Best performing models for each training strategy are **highlighted**. Values denoted with – represent not performed experiments. Values which are sourced from a graph are subject to human read-out error.

Information				Number Labeled Training Samples			
Paper	Training	Model	Source	500	1000	2000	2500
Chan et al.	ST	WRN28-2	Table	-	-	-	33.2
Mittal et al.	ST	WRN28-2	Graph	9	12	24	27
TA-VAAL	ST	ResNet18	Graph	-	-	20	-
Bengar et al.	ST	ResNet18	Graph	9	12	17	-
Ours	ST	ResNet18	Table	**14.0**	**22.4**	**32.0**	**36.3**
Chan et al.	Self-SL	WRN28-2	Table	-	-	-	49.1
Bengar et al.	Self-SL	ResNet18	Table	**47**	**50**	**56**	-
Ours	Self-SL	ResNet18	Table	37.3	45.2	52.2	54.7
Chan et al.	Semi-SL	WRN28-2	Table	-	-	-	**67.6**
Mittal et al.	Semi-SL	WRN28-2	Graph	26	35.5	44.5	49
Ours	Semi-SL	ResNet18	Graph	**41**	-	**56.5**	-

A.8 Detailed limitations

Additionally to the limitations already discussed in section 5.3.2 we would like to critically reflect on the following points:

Query methods We only evaluate four different QMs which is only a small sub-selection of all the QMs proposed in the literature. We argue that this may not be optimal, however, deem it justified due to the variety of other factors which we evaluated. Further, we excluded all QMs which induce changes in the classifier (s.a. LLAL Yoo and Kweon, 2019b) or add a substantial additional computational cost by training new components (s.a. VAAL Sinha et al., 2019). These QMs might induce changes in the HPs for every dataset and were therefore deemed too costly to properly optimize.
We leave a combination of P4 with these QMs for future research.

Validation set size The potential shortcomings of our validation set were already discussed. However, we would like to point out that a principled inclusion of K-Fold Cross-Validation into AL might alleviate this problem. This would also give direct access to ensembles which have been shown numerous times to be beneficial with regard to final performance (also in AL) Beluch et al., 2018. How this would allow us to assess performance gains in practice and also make use of improved techniques for performance evaluation s.a. Active Testing Kossen et al., 2021 in the same way as our proposed solution shown in fig. A.1 is not clear to us. Therefore we leave this point up for future research.

Performance of ST models On the imbalanced datasets, the performance of our models is not steadily increasing for more samples which can be traced back to sub-optimal HP selection according to Munjal et al., 2022a. We believe that our approach of simplified HP tuning improves over the state-of-the-art in AL showcased by the superior performance of our models on CIFAR-10 and CIFAR-100. However, regularly re-optimizing HPs might be an alternative solution.

Performance of Self-SL models Our Self-SL models are outperformed on the low-Label Regime on CIFAR-100 by the Self-SL models by Bengar et al., 2021, whereas on the medium- and high-Label Regime our Self-SL models outperform them. We believe that this might be due to our fine-tuning schedule and the possibility that Sim-Siam improves over SimCLR on CIFAR-100. Since our Self-Sl models still outperform most Semi-SL models in the literature we believe that drawing conclusions from our results is still feasible. An interesting research direction would be to make better use of the Self-SL representations s.a. improved fine-tuning regimes Kumar et al., 2022.

No Bayesian Query Methods for Semi-SL The Semi-SL models were neither combined with BALD nor BatchBALD as query functions, even though we showed that small query sizes and BatchBALD can counteract the cold-start problem. Further our Semi-SL models had bigger query sizes by a factor of three, possibly additionally hindering performance gains obtainable with AL. However, in previous experiments with FixMatch, we were not able to combine it with Dropout whilst keeping the performance of models without dropout. This clearly might have been an oversight by us, but we would like to point out that in the works focusing on AL, using Semi-SL without bayesian QMs is common practice Mittal, Tatarchenko, et al., 2019a; Gao, Z. Zhang, Yu, Arık, et al., 2020

Changing both starting budget and query size We correlated the two parameters (smaller query size for smaller starting budget etc.) since 1) in practice, we deem the size of the starting budget to be dependent on labeling cost (therefore, large query sizes for small starting budgets are unrealistic and vice versa) and 2) In this work, we are especially interested in smaller starting budgets ("cold-start" territory) compared to the ones in the literature, since AL typically shows robust performance for larger starting budgets. Theory shows that our adapted smaller query size for this case can only positively affect the result Gal, Islam, et al., 2017a; Kirsch, van Amersfoort, et al., 2019a. The only possible confounder could be that we interpret the performance of a small starting budget too positively due to a hidden effect of the smaller query size. However, we performed the low-label query size ablation, showcasing that varying the query size for small starting budgets did not have considerable effects on performance for all QMs, except BALD, where, a clear performance increase for smaller query sizes was observed.

Table A.26: Ablation study on the performance-dip on MIO-TCD and ISIC-2019 for ST models with regard to HP. Reported as mean (std).

Dataset	Labeled Train Set	Data Augmentation	Learning Rate	Weight Decay	Balanced Accuracy (Val)	Balanced Accuracy (Test)
ISIC-2019	7200	RandAugmentMC (ISIC)	0.1	5E-3	54.4(1.2)	52.6(1.9)
				5E-4	57.2(1.7)	55.4(0.9)
			0.01	5E-3	58.0(1.7)	55.6(1.0)
				5E-4	55.6(1.9)	54.5(2.1)
MIO-TCD	7700	RandAugmentMC (ImageNet)	0.1	5E-3	65.7(1.8)	64.3(1.1)
				5E-4	65.9(2.4)	63.6(2.7)
			0.01	5E-3	64.1(1.2)	62.9(1.0)
				5E-4	63.8(0.7)	62.2(1.1)

A.8.1 Instability of hyperparameters for class imbalanced datasets

The substantial dip in performance on MIO-TCD and ISIC-2019 for approx 7k samples is ablated in table A.26 where we show that simply changing the learning rate leads to stabilizing the performance on both datasets for these cases.

However, this dip in performance also arises using weighted Cross-Entropy (weighted CE-Loss) as a loss function as shown in the following ablation fig. A.5.

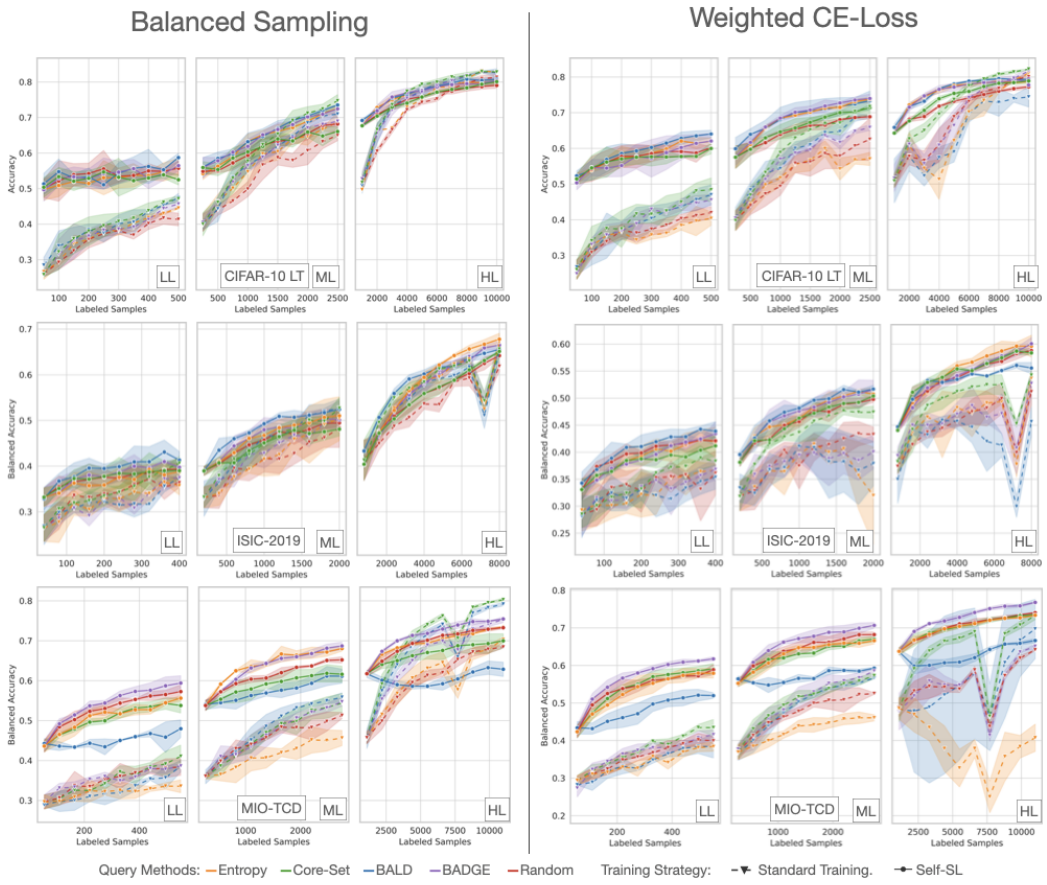


Figure A.5: Comparison between balanced sampling and weighted cross-entropy-loss (weighted CE-Loss). Whereas the ST models overall seem to benefit more from balanced sampling, the Self-SL models perform slightly better for weighted CE-Loss. Generally the observed performance gains in the imbalanced settings are still present.

Appendix B

Principled Evaluation of Uncertainties for Semantic Segmentation

This chapter of the Appendix is based on the appendix of Kim-Celine Kahl, Carsten T. Lüth, Maximilian Zenk, Klaus Maier-Hein, and Paul F Jaeger (2024b). "ValUES: A Framework for Systematic Validation of Uncertainty Estimation in Semantic Segmentation". In: *The Twelfth International Conference on Learning Representations*.

B.1 Downstream Tasks & Metrics

Here's the rephrased text with LaTeX formatting intact:

"'latex

B.1.1 Segmentation Performance Assessment

Dice To assess the segmentation performance of both the segmentation backbone and prediction models, we utilize the Dice score, defined as:

$$\text{Dice}(\hat{y}, y^*) = \frac{2|y^* \cap \hat{y}|}{|y^*| + |\hat{y}|} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{B.1}$$

Given that most prediction models generate multiple segmentation predictions $\hat{y}$ and we have multiple reference segmentations y^*, we compute the average Dice score between each of the N reference segmentations and the mean prediction $\bar{y}$:

$$\text{Dice} = \frac{1}{N} \sum_{i=1}^{N} \text{Dice}(\bar{y}, y_i^*) \tag{B.2}$$

B.1.2 Out of Distribution Detection

Our evaluation focuses on image-level OoD-D to enable practical human assessment, since humans naturally evaluate entire images rather than isolated pixels. Importantly, when any portion of an image is classified as OoD, it can affect all predictions within that image, compromising their reliability.

Area Under the Receiver Operating Characteristics Curve (AUROC) We compute the Area Under the Receiver Operating Characteristics Curve (AUROC) to evaluate each method's

ability to detect OoD cases. We assign binary labels to each image: 1 for OoD images and 0 for i.i.d. images. The sklearn library is employed to generate the ROC curve[1] using ground truth labels (0 or 1) and uncertainty scores as inputs. Subsequently, we calculate the AUC using sklearn[2].

B.1.3 Failure Detection

Our evaluation emphasizes image-level FD to support practical human assessment, as humans naturally evaluate complete images rather than individual pixels. Accordingly, we leverage our image-level performance metric (Dice) to establish a continuous failure label for our FD metrics.

Our rationale for this approach stems from the fact that Dice-based FD provides more meaningful insights into model performance than pixel-level analysis, since determining whether a human should review an image instead of relying on automated decision-making requires an assessment of overall segmentation quality rather than individual pixel quality.

We calculate our FD metrics twice: initially on i.i.d. test data, then on OoD test data. This dual evaluation allows us to determine how effectively failures are identified within i.i.d. data and, subsequently, how well the uncertainty method detects failures when encountering OoD data.

Area under the Risk-Coverage-Curve (AURC) The Area under the Risk-Coverage-Curve (AURC) serves as a metric for selective classification. The objective is to successfully identify failures by maintaining low *risk* (indicating strong classifier performance) while achieving high *coverage* (minimizing cases requiring manual correction). To calculate the Area under the Risk-Coverage-Curve, we employ the implementation from Paul F Jaeger et al., 2023. For adaptation to a semantic segmentation predictor f and evaluation dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$, we define the *confidence scoring function* (CSF) $g(x_i)$ as the negated uncertainty score. Additionally, we select the inverted Dice score as the risk l associated with each prediction:

$$l(x, y, f) = 1 - \text{Dice}(f(x), y) \tag{B.3}$$

The risk-coverage curve is constructed by introducing a confidence threshold τ, yielding the selective risk

$$\text{Risk}(\tau | f, g, D) = \frac{\sum_{i=1}^{N} l(x_i, y_i, f) \cdot \mathbb{I}(g(x_i) \geq \tau)}{\sum_{i=1}^{N} \mathbb{I}(g(x_i) \geq \tau)} \tag{B.4}$$

and coverage, defined in Paul F Jaeger et al., 2023 as the proportion of cases retained after selection:

$$\text{Coverage}(\tau | g, D) = \frac{\sum_{i=1}^{N} \mathbb{I}(g(x_i) \geq \tau)}{N} \tag{B.5}$$

The AURC based on a threshold list $\{\tau\}_{t=1}^{T}$ with T ascending-sorted CSF values can then be computed as Paul F Jaeger et al., 2023:

$$\text{AURC}(f, g, D) = \sum_{t=1}^{T} (\text{Coverage}(\tau_t) - \text{Coverage}(\tau_{(t-1)})) \cdot (\text{Risk}(\tau_t) + \text{Risk}(\tau_{t-1}))/2 \tag{B.6}$$

where we omit the conditioning on f, g, D on the right-hand side for clarity.

Excess-AURC (E-AURC) Furthermore, following the analysis in Paul F Jaeger et al., 2023 and originally proposed in Geifman et al., 2019, we employ the *excess AURC* (E-AURC) as an evaluation metric that operates independently of the segmentation model's performance:

$$\text{E-AURC} = \text{AURC}(f, g, D) - \text{AURC}(f, g^*, D) \tag{B.7}$$

where the second term represents the optimal AURC. The optimal CSF g^* can be formally derived, for instance, by employing an oracle CSF that assigns confidence equal to the negative risk of a specific prediction, $g^*(x) = -l(x, y, f)$. In practice, this perfectly ranks predictions by their risk (in

[1] `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_curve.html`
[2] `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.auc.html`

our case, ascending Dice scores). While we recognize that evaluating a CSF without considering the model's performance itself limits meaningful comparison of uncertainty methods (see Paul F Jaeger et al., 2023), we utilize this as a supplementary diagnostic metric, which is viable in our case since no significant outliers exist in segmentation performance, as evidenced by the Dice scores in table B.5.

B.1.4 Active Learning

Our evaluation concentrates on image-level AL queries to enable practical human assessment, as humans naturally evaluate complete images rather than individual pixels. The fundamental concept involves a model already performing well on an i.i.d. dataset with saturated performance for a given task, which must be adapted to a shifted (OoD) dataset with the same task. Consequently, we exclusively measure performance improvement on the OoD test set.

Active Learning Improvement (AL improvement) To evaluate the AL improvement of uncertainty methods, we measure the relative performance change between two cycles t_1 and t_2 on the OoD test set:

$$C = \frac{\text{Dice}_{t_2} - \text{Dice}_{t_1}}{\text{Dice}_{t_1}} \tag{B.8}$$

To exclude the effects of random querying from our evaluation, we subtract the performance change achieved through random querying from the performance change of the uncertainty method, yielding the following adjusted performance change:

$$C_{\text{final}} = C_{\text{method}} - C_{\text{random}} \tag{B.9}$$

B.1.5 Calibration

Our CALIB evaluation follows standard protocol, performed with pixel-level ground truth and aggregated to individual images, thus requiring no additional aggregation.

We compute our CALIB metrics twice: first on i.i.d. test data, then on OoD test data. This dual approach allows us to assess how well the uncertainty measure is calibrated on i.i.d. data and, subsequently, how well it maintains calibration when exposed to OoD inputs.

Average Calibration Error (ACE) The Average Calibration Error (ACE) is introduced in Neumann et al., n.d. and applied to segmentation in Jungo et al., 2020. Unlike the Expected Calibration Error (ECE), used in works such as Gustafsson et al., 2020; Jungo et al., 2020, ACE weights each bin in the calibration histogram equally, resulting in the following formulation:

$$\text{ACE} = \frac{1}{M} \sum_{m}^{M} |c_m - \text{Acc}_m| \tag{B.10}$$

Here, M represents the number of non-empty bins, c_m denotes the average confidence in bin m, and Acc_m indicates the corresponding average accuracy. We apply Platt scaling to obtain confidence scores ranging between 0 and 1. We selected this metric over ECE as it prevents overweighting background pixels, which predominate in our application.

B.1.6 Ambiguity Modeling

Our AM evaluation comprises two primary components: first, determining whether an uncertainty measure can successfully indicate AU in appropriate regions, and second, assessing whether a prediction model can generate multiple plausible predictions.

The evaluation utilizes pixel-level ground truth based on individual images, thus requiring no additional aggregation.

We compute our AM metrics twice: first using exclusively i.i.d. test data, then on OoD test data. This dual approach enables us to evaluate how effectively the uncertainty measures model AU on i.i.d. data and, subsequently, their performance on OoD data.

Normalized Cross-Correlation (NCC) We compute the normalized cross-correlation (NCC) following S. Hu et al., 2019:

$$\frac{1}{n_p \sigma_a \sigma_b} \sum_{i=1}^{n_p} (a_i - \mu_a) \cdot (b_i - \mu_b) \tag{B.11}$$

Here, a represents the reference uncertainty map, b denotes the predicted uncertainty map, n_p indicates the total pixel count in the uncertainty maps, and μ and σ represent the mean and standard deviation of the uncertainty maps. The reference uncertainty map is calculated using the pixel variance of pixel y_i across N different segmentation raters $\{y_i^1, ..., y_i^N\}$:

$$\mathbb{V}_{p(D)}[y_i] = \frac{1}{N} \sum_{j=1}^{N} (y_i^j - \bar{y}_i)^2 \tag{B.12}$$

where $\bar{y}_i$ represents the mean across segmentation raters $\bar{y}_i = \frac{1}{N} \sum_{j=1}^{N} y_i^j$.

Generalized Energy Distance (GED) To better evaluate the capability of uncertainty methods to model multiple raters, we employ the generalized energy distance (GED), which has been utilized in numerous works focusing on AM (S. Kohl et al., 2018; Monteiro et al., 2020; S. Hu et al., 2019):

$$D^2_{\text{GED}}(p, \hat{p}) = 2\mathbb{E}_{y\sim p, \hat{y}\sim\hat{p}}[d(y, \hat{y})] - \mathbb{E}_{y,y'\sim p}[d(y, y')] - \mathbb{E}_{\hat{y},\hat{y}'\sim\hat{p}}[d(\hat{y}, \hat{y}')] \tag{B.13}$$

Here, $d(y, y')$ measures the distance between two reference segmentations, while $d(\hat{y}, \hat{y}')$ quantifies the distance between two predicted segmentation variants. p and $\hat{p}$ represent the respective reference and predicted distributions for segmentation masks. The distance metric must satisfy two conditions: it increases with greater mask dissimilarity and $d(x, y) = 0$ when $x = y$. Since we employ Dice as our primary evaluation metric, we define $d(x, y) = 1 - \text{Dice}(x, y)$ as our distance measure. “‘

B.2 Datasets

B.2.1 Toy dataset setup

Dataset scenarios

As outlined in section 6.4.3, we construct three distinct training scenarios and four distinct testing scenarios for the toy dataset. table B.1 provides an overview of these scenarios, including the number of training and testing cases within each. Each scenario is designed to address specific questions in our separation study, as described in section 6.4.1. Setting 1 introduces AU, thereby targeting Q1 and Q2 of the separation study. Setting 2 emphasizes EU, consequently addressing Q3 and Q4. However, given that AU is absent in setting 2, we anticipate that AU-measure behavior will be difficult to predict, constraining our ability to definitively answer Q4. Accordingly, we develop setting 3, which provides testing scenarios (a) and (b) where AU is introduced during training, with scenario (b) additionally incorporating AU in the i.i.d. testing data. These scenarios aim to elucidate our uncertainty measures' behavior in detecting EU under varying AU conditions.

Data with induced aleatoric uncertainty

fig. B.1 illustrates the data scenario generated with induced aleatoric uncertainty. The input (fig. B.1a) depicts a sphere with Gaussian blur extending outward. This outward blur creates ambiguity regarding the sphere's precise boundary. Three different reference raters model this ambiguity (fig. B.1b - fig. B.1d). Specifically, rater 1's segmentation (fig. B.1b), which delineates the smallest sphere, measures 10% of rater 3's segmentation size (fig. B.1d). Rater 2 (fig. B.1c)

Table B.1: Number of training and testing cases for the toy dataset. For each scenario, the number of training cases and the number of testing cases is specified. Further, the number of cases with ambiguity / blur is specified in brackets and the number of i.i.d and OoD cases in the testset.

Scenario	Description	# Train (# blur)	# Test	
			# i.i.d (# blur)	# OoD
1	Training models on data with induced AU; testing on i.i.d. data also containing AU	200 (200)	20	
			20 (20)	0
2	Training models on data without ambiguity; testing on i.i.d. data and shifted data	200 (0)	42	
			21 (0)	21
3a	Training models on data with and without blur/ambiguity; testing on i.i.d. data and shifted data without blur	200 (100)	42	
			21 (0)	21
3b	Training models on data with and without blur/ambiguity; testing on i.i.d. data and shifted data without blur and i.i.d data with blur	200 (100)	63	
			42 (21)	21

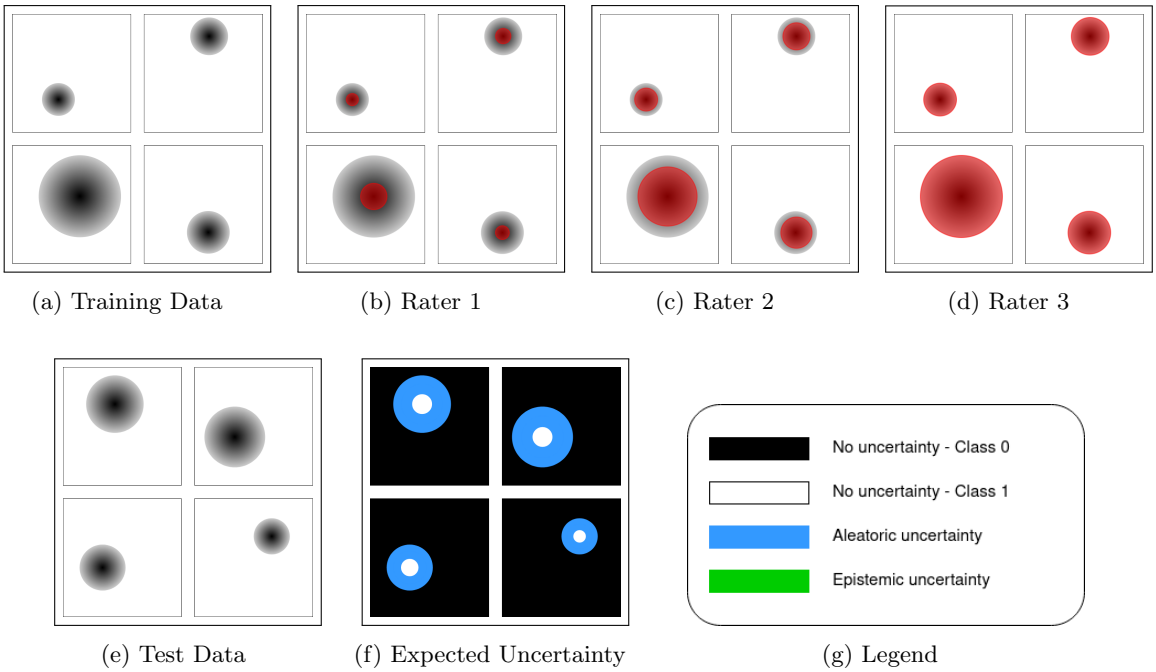
occupies the midpoint between these extremes, with its segmentation measuring 55% of rater 3's size.

The test set (fig. B.1e) follows the identical construction procedure as the training set. fig. B.1f displays the anticipated uncertainty, with the corresponding legend in fig. B.1g. Upon model convergence after training, epistemic uncertainty should be absent from the data since the test set replicates the training set's construction. Instead, only aleatoric uncertainty should manifest, concentrated in the ambiguous boundary region of the sphere.

Data with induced epistemic uncertainty

fig. B.2 presents the data scenario constructed with induced epistemic uncertainty. The training data input object (fig. B.2a) consists of a sphere, similar to the aleatoric uncertainty dataset. However, in the epistemic data scenario, this sphere lacks outward blur, establishing a well-defined segmentation boundary for the ground truth segmentation (fig. B.2b). Since a purely black background would oversimplify the segmentation task, random noise is incorporated into the background. The test set for this dataset appears in fig. B.2c. It comprises objects with diverse shapes and colors absent from the training data. While some objects remain spherical with varying gray values, the test set also includes cubes and spheres partially extending beyond the image boundaries, whereas training set spheres were invariably contained within the image.

Given that all segmentations are unique, aleatoric uncertainty should be absent from the data. However, the expected location of epistemic uncertainty remains less clear. Network generalization capabilities may vary depending on which features were predominantly learned during training (Geirhos et al., 2020). For this particular toy example, the network's learned training solution remains uncertain. If the network learned shape recognition, novel shapes should elicit elevated prediction uncertainty, as illustrated in fig. B.2d. Conversely, if intensity learning predominated, epistemic uncertainty might resemble fig. B.2e. Alternatively, the network may have learned a different decision rule, potentially yielding distinct epistemic uncertainty patterns.



(a) Training Data (b) Rater 1 (c) Rater 2 (d) Rater 3

(e) Test Data (f) Expected Uncertainty (g) Legend

Figure B.1: Aleatoric data scenario. (a) shows the input images in the training set, which are ambiguous due to Gaussian blur to the outside. (b) - (d) show three different reference ratings that are generated for the input images. (e) shows test images and (f) the expected uncertainty maps. The uncertainty regions are explained in (g).

B.2.2 LIDC-IDRI dataset setup

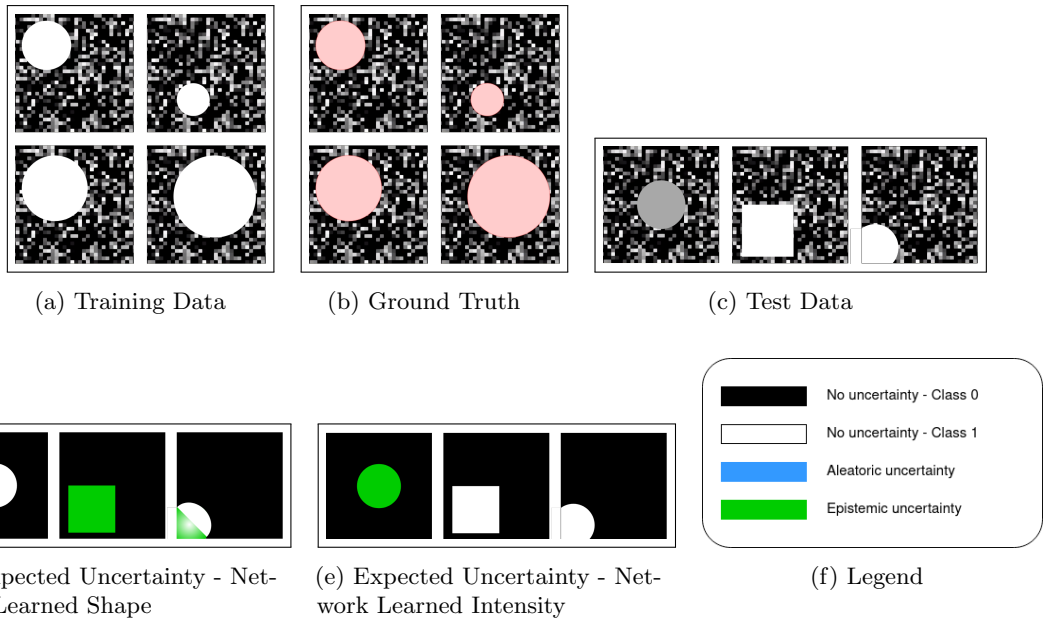
Dataset preprocessing

For dataset preprocessing, we employ the pylidc library (Hancock and Magnan, 2016). This library enables querying and clustering of all nodules measuring ≥ 3 mm, ensuring each nodule receives annotations from up to four raters. We exclude cases positioned too closely together for automatic grouping into a single nodule. Additionally, we compute a consensus mask representing the union of all rater annotations and discard cases where this consensus mask exceeds 64 voxels along any dimension. We extract patches of size $64 \times 64 \times 64$ centered on each nodule, with all images resampled to a uniform resolution of $1 \times 1 \times 1$ mm. Our subsequent analysis focuses exclusively on nodules annotated by all four raters, totaling 901 nodules.

Metadata distribution shift analysis

The dataset encompasses nine distinct metadata features: *subtlety*, *internal structure*, *calcification*, *sphericity*, *margin*, *lobulation*, *spiculation*, *texture* and *malignancy*. Each feature contains 4-6 possible categories, with segmentation raters assigning one category per feature. To induce distribution shifts, we convert each metadata feature from its original categories into binary classes (in-distribution and out-of-distribution). To minimize confounding with aleatoric uncertainty in our distribution shift analysis, we exclude *subtlety* and *margin*, since subtle nodules may lack complete rater annotations and indistinct margins can introduce high boundary variability. We also omit *internal structure* due to having only a single OoD case, rendering it unsuitable for meaningful i.i.d./OoD comparison.

Subsequently, we establish a train/test partition to examine the performance differential of a deterministic U-Net model between i.i.d. and OoD test sets. fig. B.3 illustrates this partitioning strategy. Initially, we eliminate nodules lacking a majority vote classification (i.e., cases with two raters voting i.i.d. and two voting OoD). We then identify all patients containing at least one OoD nodule. Their OoD nodules populate the OoD test set, while their i.i.d. nodules comprise the i.i.d. test set. For remaining patients with exclusively i.i.d. nodules, the majority are allocated to the i.i.d. training set, with a subset assigned to the i.i.d. test set to achieve an 80%/20% training/test ratio.



(a) Training Data (b) Ground Truth (c) Test Data

(d) Expected Uncertainty - Network Learned Shape (e) Expected Uncertainty - Network Learned Intensity (f) Legend

Figure B.2: Epistemic data scenario. (a) shows the input images in the training set. (b) shows the ground truth segmentation. (c) shows test images that differ in various aspects from the training data. (d) and (e) show possible uncertainty maps, depending on what the network learned. The uncertainty regions are explained in (f).

Patient identifiers determine this allocation between training and i.i.d. test sets. This partitioning strategy guarantees patient-level separation, preventing any patient's nodules from appearing in both training and test sets simultaneously.

To quantify the performance degradation between i.i.d. and OoD test sets, we train 5 folds per metadata split with different random seeds across folds. We calculate the mean Dice coefficient (comparing predictions against one randomly selected rater) along with standard deviations for both i.i.d. and OoD test sets to assess performance. table B.2 presents these results.

Following performance drop assessment across all features, we select the two features exhibiting the most substantial performance degradation for further experimentation. These are the texture shift and malignancy shift. The results reveal a considerable performance decline between i.i.d. and OoD conditions, validating our approach for inducing epistemic uncertainty.

Table B.2: Results for the LIDC-IDRI shift analysis. For each feature, the Dice score on the i.i.d. test set, the Dice score on the OoD test set and the performance drop between i.i.d. and OoD test set are shown. Mean and standard deviation are reported for training with 5 folds, each with a different seed.

Feature	i.i.d. / OoD	Dice i.i.d.	Dice OoD	Performance Drop (%)
Calcification	Absent / Present	0.804 ± 0.0022	0.7669 ± 0.0133	4.6112 ± 1.7699
Sphericity	Round / Linear	0.7934 ± 0.0042	0.7474 ± 0.0106	5.7905 ± 1.191
Lobulation	No Lobulation / Lobulation	0.7887 ± 0.0042	0.7649 ± 0.0046	3.0163 ± 0.6264
Spiculation	No Spiculation / Spiculation	0.7958 ± 0.0022	0.7458 ± 0.0068	6.2865 ± 0.9447
Texture	Solid & Part Solid / Non-solid	0.81 ± 0.0012	0.6081 ± 0.0124	24.9244 ± 1.431
Malignancy	Non-malignant / Malignant	0.7789 ± 0.0051	0.6677 ± 0.0645	14.3093 ± 7.9522

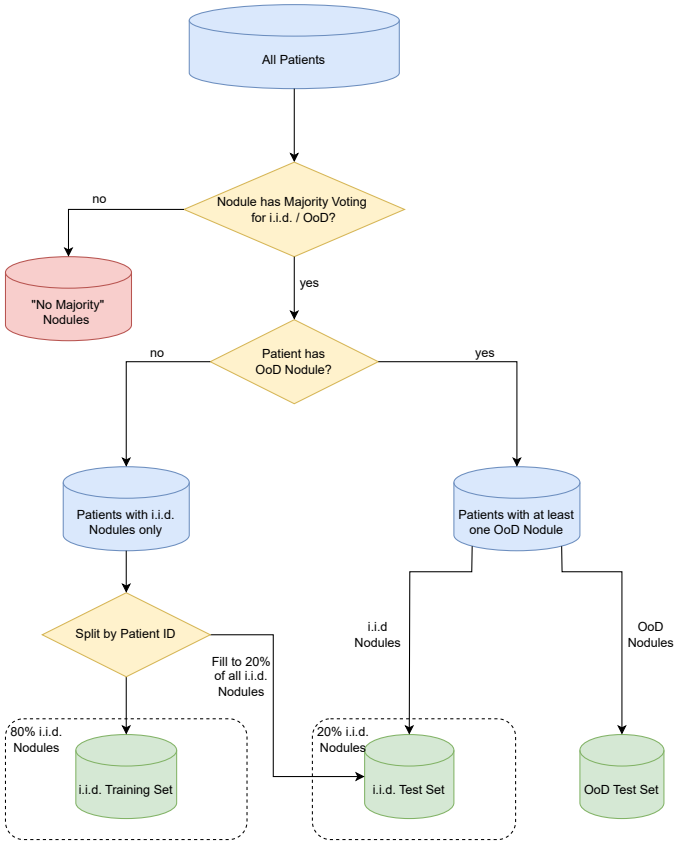


Figure B.3: Splits for the LIDC-IDRI shift analysis. Only nodules are considered that have a majority vote for either being i.i.d. or OoD. Furthermore, the splits are created considering the patient ID, so that no nodules of the same patient are in the training set and the test set at the same time. In the end, an i.i.d. training set, an i.i.d. test set, and an OoD test set are created to analyze the shifts between the features.

Setup for evaluation on downstream tasks

To assess performance across various downstream tasks, we partition lung nodules into three subsets: i.i.d. training set, i.i.d. and OoD test sets, and i.i.d. and OoD unlabeled pools. table B.3 displays the sizes of these subsets. We begin by training the model exclusively on the i.i.d. training set, assuming performance saturation on i.i.d. data. Following this initial training, we evaluate uncertainty method performance on FD, CALIB, and AM. We then rank samples from the unlabeled pool by uncertainty scores. Using these uncertainty rankings, we assess each method's capability for OoD sample detection and augment the training pool with the top 50% most uncertain samples, targeting improved OoD test set performance. With this expanded training set, we conduct another training iteration and subsequently re-evaluate test set performance.

Table B.3: Size of the different sets in the LIDC dataset for the evaluation on the various downstream tasks.

Split	Train	Val	Test		Unlabeled Pool	
			i.i.d	OoD	i.i.d	OoD
Texture	513	129	167	20	42	20
Malignancy	200	51	105	93	184	92

B.2.3 GTA5/Cityscapes dataset setup

As an additional dataset, we employ a pairing of the GTA5 dataset (Richter et al., 2016) with the Cityscapes dataset (Cordts et al., 2016a). As discussed in section 6.4.3, these datasets share identical class labels, allowing us to treat GTA5 as in-distribution data while Cityscapes serves as out-of-distribution data. For the Cityscapes dataset, we designate the training split as an unlabeled pool for active learning downstream tasks, while the validation split functions as our test set. fig. B.4 illustrates the dataset partitioning scheme along with the specific image counts for each subset. Specifically, we randomly sample an equivalent number of images from GTA5 to form its unlabeled pool and establish a 75/25 train/test division for the GTA5 dataset. While the

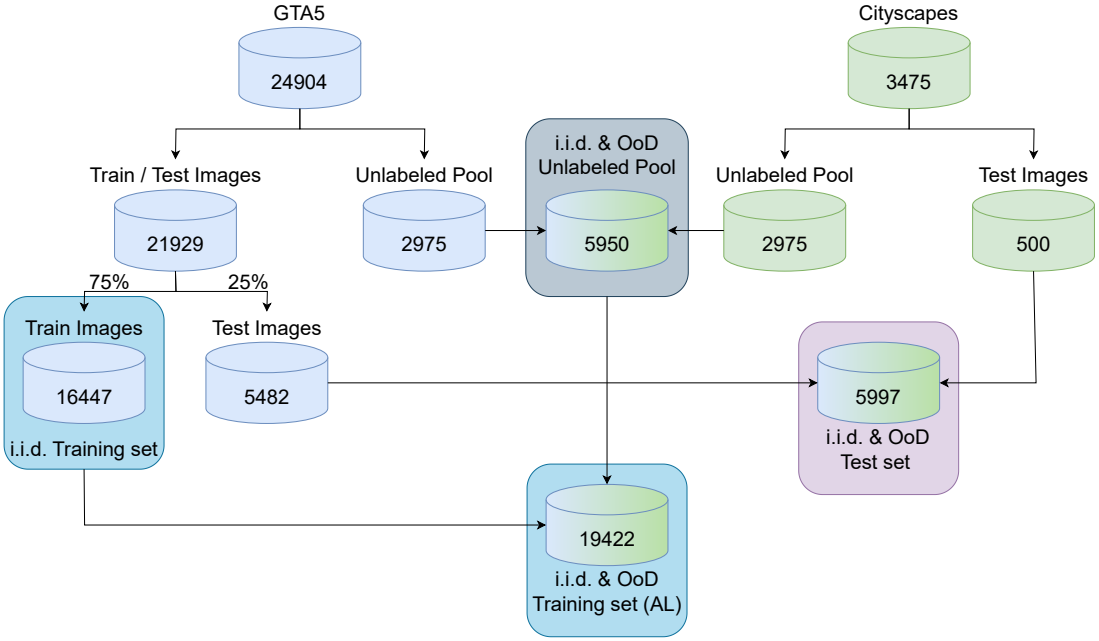


Figure B.4: Splits for the GTA5/CS dataset. From the Cityscapes dataset, the training set is used as unlabeled pool and the validation set is used as test set.

Cityscapes dataset encompasses up to 30 classes, only 19 are utilized during validation. Since the GTA5 dataset contains exclusively these 19 classes, we limit our analysis accordingly. Additionally, as noted in section 6.4.3, we implement random class transitions for the categories "sidewalk", "person", "car", "vegetation" and "road" with a probability of $\frac{1}{3}$ from <class> to <class 2>. This methodology follows the approach adopted by S. Kohl et al., 2018. We initially crop all images to dimensions of 1024×1912, then downscale them to 25% of their original size, yielding final image dimensions of 256×478.

B.3 Model implementation details

In this section, we detail the implementation specifications for the model backbones and prediction models. We note that extensive hyperparameter optimization was not conducted for all parameters listed here; instead, we employed standard values when they yielded reasonable performance, and performed targeted hyperparameter sweeps only when necessary to identify suitable configurations. We acknowledge that the prediction model implementations involve numerous hyperparameters, and more comprehensive hyperparameter exploration could represent a valuable direction for future investigation. All models undergo training for 150 epochs.

B.3.1 Segmentation Backbones

U-Net For both the toy dataset and LIDC datasets, we employ a 3D U-Net architecture as the segmentation backbone. The initial filter size is set to 8 for the toy dataset and 16 for the LIDC

datasets, with four encoder blocks and four decoder blocks. Our loss function combines Dice loss and cross-entropy loss, with the exception of SSNs as the prediction model. For SSNs, we adopt the loss function specified in Monteiro et al., 2020. We utilize the Adam optimizer with a learning rate of $3e-4$ and weight decay of $1e-5$. The batch size is configured at 8. Data augmentation includes random flipping and Gaussian noise.

HRNet For the GTA5/CS dataset, we implement the HRNet as our segmentation backbone, utilizing ImageNet pretrained weights. The cross-entropy loss serves as our loss function, once again with the exception of SSNs. For all prediction models excluding SSNs, we employ SGD as the optimizer with a learning rate of 0.01, weight decay of $5e-4$, and momentum of 0.9. For SSNs specifically, RMSprop serves as the optimizer with a learning rate of $1e-4$, weight decay of $5e-4$, and momentum of 0.6. The batch size is set to 6. Our augmentation strategy encompasses random horizontal flipping, rotations, random scaling, random cropping, and Gaussian noise.

B.3.2 Prediction Models

Test-time dropout (TTD) For the U-Net architecture, we insert dropout layers after each convolutional block with probability $p = 0.5$. For the HRNet architecture, we position dropout layers at the terminus of each branch, following the approach in Nash et al., 2022. The probability remains $p = 0.5$. At inference time, we execute 10 MC-Dropout forward passes for each input.

Ensemble For ensemble models, the model architectures and training procedures remain unchanged; we simply train 5 models using different random seeds. During inference, each input image is processed through all 5 models.

Test-time data augmentations (TTA) For TTA models, we apply identical augmentations to those used during training for the 3D U-Net. Specifically, we generate all possible combinations of flipping operations and Gaussian noise, yielding 16 forward passes per input image (8 potential flipping directions, each with or without noise). For the HRNet, we similarly apply all combinations of random horizontal flipping and Gaussian noise, producing 4 forward passes per input image (2 flipping options, each with or without noise).

Stochastic Segmentation Networks (SSNs) For stochastic segmentation networks, we perform 10 forward passes per input image. We employ a rank of 5 for the toy dataset and LIDC datasets, while using a rank of 10 for the GTA5/CS dataset. Since training stability improved when pretraining the mean component first, we conduct 5 pretraining epochs where only the mean is trained before incorporating covariance matrix training.

B.4 Uncertainty Measures for Probabilistic Variability Variable Prediction Models

For a probabilistic prediction model $p(Y|x) = \mathbb{E}_{z\sim p(z)}[p(Y|x,z)]$ that predicts the class variable Y for a given sample x using an additional variable Z following $p(z)$ intended to capture rater/label variability (variability variable), we propose that AU and EU can be estimated analogously to Bayesian models, following

$$\underbrace{H[Y|x]}_{\text{PU}} = \underbrace{I[Y;Z|x]}_{\text{AU (for i.i.d. } x)} + \underbrace{\mathbb{E}_{z\sim Z}[H(Y|z,x)]}_{\text{EU}}. \tag{B.14}$$

Representative examples of these approaches include SSNs (Monteiro et al., 2020), the probabilistic U-Net (S. Kohl et al., 2018), and PHiSeg (C. F. Baumgartner et al., 2019), where the prediction model is explicitly trained to learn rater variability.
A more comprehensive rationale is presented in the following two paragraphs, with the third paragraph describing the reason for our observed failure mode.

Aleatoric uncertainty. Multiple plausible predictions for a sample arising from ambiguity or other factors are traditionally classified as AU (Monteiro et al., 2020; A. Kendall and Gal, 2017b), leading to the assumption that the variability variable Z fundamentally captures the prediction model's learned AU. Consequently, the mutual information between the class label Y and the variability variable Z conditioned on a sample x quantifies how much information about AU could be obtained by acquiring the class label y.

$$I[Y; Z|x] = H[Y|x] - \mathbb{E}_{z\sim Z}[H[Y|x, z] \tag{B.15}$$

Knowledge of the optimal variability variable Z would effectively reduce the uncertainty. We therefore hypothesize that this uncertainty measure captures AU.

Epistemic uncertainty. Following the principle that a variability variable prediction model should never exhibit uncertainty about its predictions on i.i.d. data when still dependent on the variability variable $p(Y|x, z)$ [3] Therefore, the classifier's uncertainty $H[Y|x]$ that cannot be attributed to the variability variable Z should represent novel and previously unseen patterns (from the prediction model's perspective). Following this logic, we hypothesize that the expected entropy of the variability variable captures EU.

$$\mathbb{E}_{z\sim Z}[H[Y|x, z]] = H[Y|x] - I[Y; Z|x] \tag{B.16}$$

Failure mode. In our experiments with SSNs, we observe that while the model remains dependent on the variability variable, uncertainty typically concentrates in border regions between classes but rarely extends to larger image regions [4]. This provides an explanation for why in the LIDC-IDRI dataset experiments, where rater disagreement predominantly occurs in nodule border regions for most samples, $I[Y; Z|x]$ achieves the lowest NCC scores (Q1 + Q2).

B.5 Uncertainty Measures for Test-Time Augmentation Models

Consider a model employing a set of label-preserving data augmentations during inference, defined on the input space $\mathcal{T}$ and represented as a random variable T (support$(T) = \mathcal{T}$) from which samples $t \sim T$ are drawn. Inference utilizing test-time augmentations can be expressed as $p(Y|x) = \mathbb{E}_{t\sim T}[p(Y|t, x)] = \mathbb{E}_{t\sim T}[p(Y|t(x))]$. During training, the model is optimized on the training set $\mathcal{D}$ using a training objective (typically cross-entropy loss (CE-Loss)) to achieve invariance to augmentations within $\mathcal{D}$. For an optimal model where the training objective is minimized (e.g., CE-Loss=0), the model's outputs on the training set $\mathcal{D}$ exhibit complete invariance to all transformations $p(Y|x, t_1) = p(Y|x, t_2) \forall t_1, t_2 \in \mathcal{T}, x \in \mathcal{D}$.
For such a model, we propose that AU and EU can be estimated analogously to Bayesian models, following

$$\underbrace{H[Y|x]}_{\text{PU}} = \underbrace{I[Y; T]}_{\text{EU}} + \underbrace{\mathbb{E}_{t\sim T}[H[Y|t, x]}_{\text{AU (for i.i.d. } x)}. \tag{B.17}$$

A more comprehensive rationale is provided in the subsequent two paragraphs.

Aleatoric uncertainty. Since our model achieves perfect training set performance, it can detect previously observed uncertainty across augmentations, analogous to how Bayesian models accomplish this through parameter sets (Mukhoti, van Amersfoort, et al., 2021). Consequently, the expected entropy across augmentations should convey information regarding the AU level in a datapoint's prediction.

$$\mathbb{E}_{t\sim T}[H[Y|x, t]] \tag{B.18}$$

[3] This design principle is inherent in the training of variability variable prediction models. For instance, in SSNs, this is implemented through the logsumexp of the logarithmic loss (Monteiro et al., 2020).

[4] We hypothesize that this behavior results from $p(z)$ modeling a Gaussian distribution in logit space.

Epistemic uncertainty. Given our model's invariance to augmentations on the training set, it follows that $I[Y;T|x] = 0 \forall x \in \mathcal{D}$ (Jensen, 1906). When the mutual information between the augmentation variable and predicted label exceeds zero ($I[Y,T|\hat{x}] > 0$) for a datapoint $\hat{x} \notin \mathcal{D}$, this signals that the datapoint deviates in some manner from $\mathcal{D}$. Moreover, if $\hat{x}$ were incorporated into the training set and the model retrained, the model would acquire invariance to augmentations for this datapoint. Following this reasoning, this term is therefore reducible through the addition of previously unseen datapoints. Based on this foundation, we hypothesize that the mutual information between the augmentation variable and the predicted label captures EU.

$$I[Y,T|x] = H[Y|x] - \mathbb{E}_{t\sim T}[H[Y|x,t]] \tag{B.19}$$

Implications. These derivations suggest that TTA actually enables the model to estimate EU rather than enhancing AU estimation. This conclusion aligns with the hypothesis proposed by S. Hu et al. (2019) and directly contradicts the assertions of two influential papers claiming it models AU (G. Wang et al., 2019; Ayhan and Berens, 2018).

B.6 Details on the aggregation strategies

B.6.1 Ablation study: Correlation of image level aggregation and object size

To validate our hypothesis regarding the relationship between object size and uncertainty levels, we created visualizations examining these two variables within the LIDC datasets. fig. B.5 presents one such visualization for a TTD model applied to the LIDC TEX dataset. The upper panels display the aggregated uncertainty versus the mean predicted segmentation size across three uncertainty types: epistemic, aleatoric, and predictive. The lower panels show the total uncertainty normalized by object size. To verify that predicted segmentation sizes align with ground truth dimensions, we plot these two measures against each other on the right side. The results reveal a positive relationship between summed uncertainty and object size in the upper panels. However, when uncertainty is averaged (lower panels), this relationship disappears. These findings indicate that total uncertainty scales with object dimensions rather than capturing size-independent uncertainty characteristics of the objects.

B.6.2 Selection of threshold for threshold level aggregation

The threshold level aggregation approach requires establishing a cutoff value to classify pixels as "uncertain." Since uncertainty typically concentrates at object boundaries, it naturally scales with object size. We therefore compute the threshold based on validation set object sizes using the following procedure: First, we calculate the mean foreground ratio α across all predicted segmentations in the validation set:

$$\alpha = \frac{\#\text{voxels foreground pred}}{\#\text{voxels}} \tag{B.20}$$

This foreground ratio determines the quantile value $q = 1 - \alpha$. We then apply this quantile to the validation set's uncertainty mapsto identify the pixel intensity Q Q at that quantile, which becomes our threshold for subsequent predictions:

$$\text{threshold} = Q(q, u_{\text{val}}) \tag{B.21}$$

This procedure yields one threshold per uncertainty modeling approach.

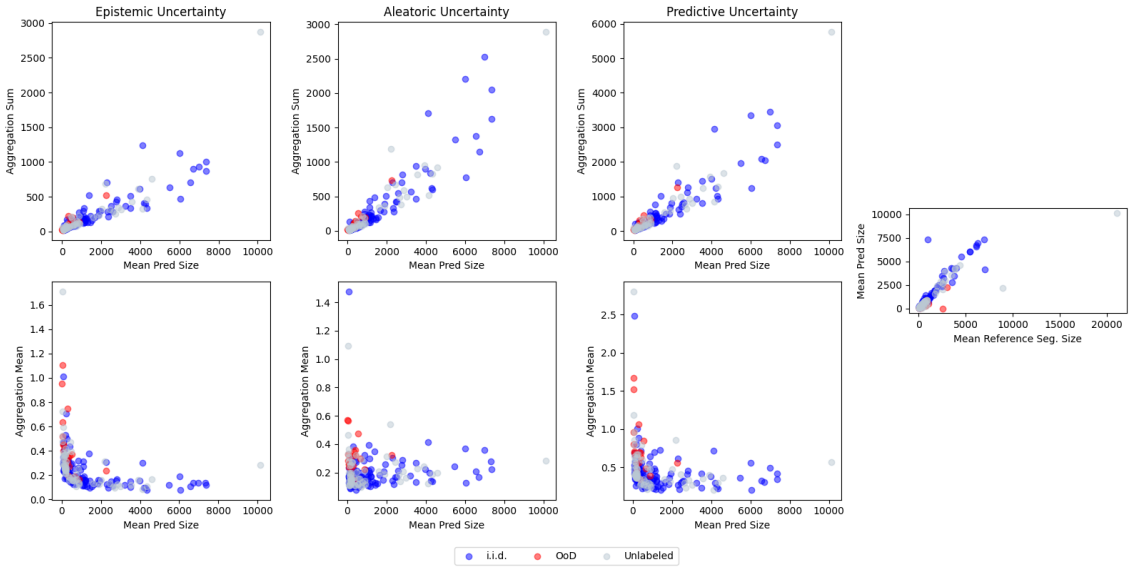


Figure B.5: Correlation between object size and uncertainty for image level aggregation. In the top row, all pixels in the uncertainty maps are added up and this aggregation sum is plotted with respect to the mean size of the predicted segmentations. In the bottom row, the aggregation sum is additionally divided by the predicted object size, resulting in the aggregation mean. On the right-hand side, the mean prediction size is plotted with respect to the mean reference segmentation size to see that the size of the predictions roughly corresponds to the reference segmentation sizes of the objects.

B.7 Detailed results of the separation study

B.7.1 Detailed analysis

Q1 & Q2 *Toy dataset.* In the toy dataset analysis, AU uncertainty measures consistently achieve higher NCC values than EU uncertainty measures, demonstrating effective separation of AU and appropriate highlighting of relevant regions (Q1). This finding is further validated by qualitative analysis showing elevated uncertainty signals in areas with rater disagreement (see section B.7.3). Conversely, EU-measures underperform relative to both PU and AU measures, suggesting that EU-measures fail to capture AU. SSNs represent an exception to this pattern, exhibiting higher NCC scores for EU-measures compared to other prediction models. This anomaly may stem from the presence of AU in border regions that is not accounted for by the variability variable.

LIDC datasets. On the LIDC datasets, EU-measures demonstrate performance comparable to AU-measures. Consequently, these approaches appear to capture EU in regions associated with AU (Q2).
Indeed, for SSNs, the AU-measure exhibits lower NCC than the EU-measure, potentially because SSNs assign high EU to border regions, which correspond to areas of disagreement. Qualitative analysis reveals that AU-measures show marginally superior AU indication when substantial inter-rater variability extends beyond narrow border regions (see fig. B.7). Notably, this phenomenon manifests only for i.i.d. nodules. For instance, a nodule demonstrating strong AU indication in the i.i.d. test set on the LIDC TEX dataset (fig. B.7) displays poor AU indication in the OoD test set on the LIDC MAL dataset (fig. B.10).

GTA5/Cityscapes dataset. For the GTA5/CS dataset, NCC scores are generally lower than those observed in other datasets. Nevertheless, AU-measures maintain at least positive correlation, whereas EU-measures often exhibit negative correlation with AU, confirming that they do not model AU effectively. SSNs represent the only prediction model achieving high AU through its corresponding AU-measure. This qualitative distinction is also evident in fig. B.11: While most prediction models display peak AU at object boundaries, the SSNs AU-measure highlights the entire ambiguous region.

Q3 & Q4 *Toy dataset.* In Setting 2, characterized by exclusive EU presence in the data, no substantial AUROC difference emerges between AU-measures and EU-measures. One might hypothesize that without AU learning opportunities in training data, all uncertainty measures effectively function as EU-measures. However, once AU is incorporated into training data in settings 3a and 3b, EU-measures become superior separators of i.i.d. and OoD data. In setting 3b, where AU exists in both training and test data, EU separation proves particularly advantageous. Regarding Q3 and Q4 on the toy dataset, EU-measures consistently achieve AUROC values exceeding random performance, confirming Q3. The response to Q4 varies depending on AU prevalence in training and test data.

LIDC datasets. On the LIDC datasets, the separation between AU and EU yields notable benefits specifically for TTD, Ensembles, and TTA. Particularly on LIDC TEX, EU-measures demonstrate superior effectiveness in separating i.i.d. and OoD data. A consistent pattern emerges: whenever PU and EU separation proves advantageous, AU as an OoD-detector performs below random baseline. An additional hypothesis emerges that EU separation appears most beneficial in scenarios where OoD-detection performance has not yet saturated, exemplified by LIDC TEX. To address Q3 and Q4 for the LIDC dataset, EU-measures consistently achieve AUROC values surpassing random performance, confirming Q3. However, Q4 receives only partial confirmation in that AU proves ineffective whenever separating EU from PU yields benefits.

GTA5/Cityscapes dataset. For the GTA5/CS dataset, most EU-measures substantially outperform their corresponding AU-measures in terms of AUROC. TTD constitutes the sole exception, where the EU-measure actually underperforms the AU-measure. Additionally, AU-measures fall below random performance for patch-level aggregation, while achieving marginally better than random performance for image-level aggregation. This collectively indicates, with respect to Q3, that EU-measures (excluding TTD) successfully capture EU, whereas for Q4, it can be largely confirmed that AU-measures fail to consistently exceed random selection for OoD case identification.

B.7.2 Quantitative results

The detailed quantitative results for the separation study, presented in section 6.4.5, can be found in table B.4. table B.4a provides insights on answering Q1 and Q2, while table B.4b addresses Q3 and Q4.

Table B.4: Quantitative results for the separation study. In order to answer Q1 and Q2 from the separation study, the NCC scores are calculated between the uncertainty maps and the variance of the reference segmentations, shown in table B.4a. To answer Q3 and Q4, the AUROC scores are calculated and reported in table B.4b. Mean results are shown over 3 runs with different seeds for all relevant dataset settings to answer the respective questions. Abbreviations: PM: Prediction model, UM: Uncertainty measure, UT: Modeled uncertainty Type (according to theory), AGG: Aggregation strategy.

Testset	PM	UM	UT	Toy 1	LIDC TEX	LIDC MAL	GTA5/CS
i.i.d	Determ.	MSR	PU	0.68	0.32	0.28	0.51
	TTD	PE	PU	0.80	0.51	0.48	0.27
		EE	AU	0.86	0.52	0.48	0.28
		MI	EU	0.47	0.46	0.45	-0.23
	Ensemble	PE	PU	0.83	0.48	0.43	0.24
		EE	AU	0.84	0.49	0.44	0.27
		MI	EU	0.51	0.39	0.36	-0.23
	TTA	PE	PU	0.82	0.46	0.41	0.25
		EE	AU	0.82	0.48	0.42	0.26
		MI	EU	0.54	0.38	0.35	-0.16
	SSN	PE	PU	0.96	0.63	0.61	0.56
		MI	AU	0.96	0.59	0.55	0.70
		EE	EU	0.80	0.64	0.62	0.05
OoD	Determ.	MSR	PU	-	0.20	0.20	0.47
	TTD	PE	PU	-	0.37	0.36	0.26
		EE	AU	-	0.37	0.39	0.26
		MI	EU	-	0.33	0.31	-0.13
	Ensemble	PE	PU	-	0.35	0.33	0.25
		EE	AU	-	0.36	0.35	0.30
		MI	EU	-	0.30	0.27	-0.06
	TTA	PE	PU	-	0.32	0.30	0.28
		EE	AU	-	0.33	0.33	0.30
		MI	EU	-	0.27	0.25	-0.04
	SSN	PE	PU	-	0.51	0.47	0.37
		MI	AU	-	0.47	0.44	0.52
		EE	EU	-	0.52	0.47	0.03

(a) NCC scores

PM	UM	UT	AGG	Toy 2	Toy 3a	Toy 3b	LIDC TEX	LIDC MAL	GTA5/CS
Determ.	MSR	PU	Patch	0.84	0.78	0.41	0.46	0.86	0.33
			Thresh	0.73	0.45	0.40	0.52	0.59	-
			Image	-	-	-	-	-	0.70
TTD	PE	PU	Patch	0.83	0.68	0.37	0.46	0.90	0.37
			Thresh	0.48	0.50	0.38	0.61	0.74	-
			Image	-	-	-	-	-	0.68
	EE	AU	Patch	0.74	0.69	0.36	0.43	0.90	0.37
			Thresh	0.53	0.53	0.29	0.40	0.88	-
			Image	-	-	-	-	-	0.68
	MI	EU	Patch	0.83	0.61	0.73	0.52	0.88	0.46
			Thresh	0.54	0.43	0.71	0.65	0.60	-
			Image	-	-	-	-	-	0.51
Ensemble	PE	PU	Patch	0.95	0.94	0.50	0.55	0.91	0.33
			Thresh	0.90	0.73	0.69	0.66	0.72	-
			Image	-	-	-	-	-	0.72
	EE	AU	Patch	0.94	0.83	0.44	0.49	0.89	0.29
			Thresh	0.78	0.19	0.12	0.53	0.53	-
			Image	-	-	-	-	-	0.67
	MI	EU	Patch	0.95	0.95	0.85	0.65	0.89	0.91
			Thresh	0.91	0.77	0.87	0.72	0.75	-
			Image	-	-	-	-	-	0.90
TTA	PE	PU	Patch	0.95	0.91	0.48	0.51	0.88	0.32
			Thresh	0.93	0.66	0.55	0.60	0.67	-
			Image	-	-	-	-	-	0.70
	EE	AU	Patch	0.95	0.83	0.44	0.46	0.87	0.29
			Thresh	0.89	0.27	0.16	0.49	0.53	-
			Image	-	-	-	-	-	0.67
	MI	EU	Patch	0.95	0.94	0.92	0.59	0.86	0.93
			Thresh	0.93	0.71	0.84	0.67	0.70	-
			Image	-	-	-	-	-	0.94
SSN	PE	PU	Patch	0.87	0.76	0.38	0.54	0.84	0.78
			Thresh	0.74	0.65	0.34	0.51	0.72	-
			Image	-	-	-	-	-	0.82
	MI	AU	Patch	0.68	0.63	0.32	0.54	0.72	0.53
			Thresh	0.67	0.51	0.25	0.49	0.57	-
			Image	-	-	-	-	-	0.55
	EE	EU	Patch	0.87	0.90	0.43	0.54	0.85	0.78
			Thresh	0.74	0.68	0.78	0.50	0.68	-
			Image	-	-	-	-	-	0.86

(b) AUROC scores

B.7.3 Qualitative results

In the following sections, samples are shown for the qualitative analysis to answer Q1 and Q2 from the separation study (see section 6.4.5).

Qualitative results for the toy dataset

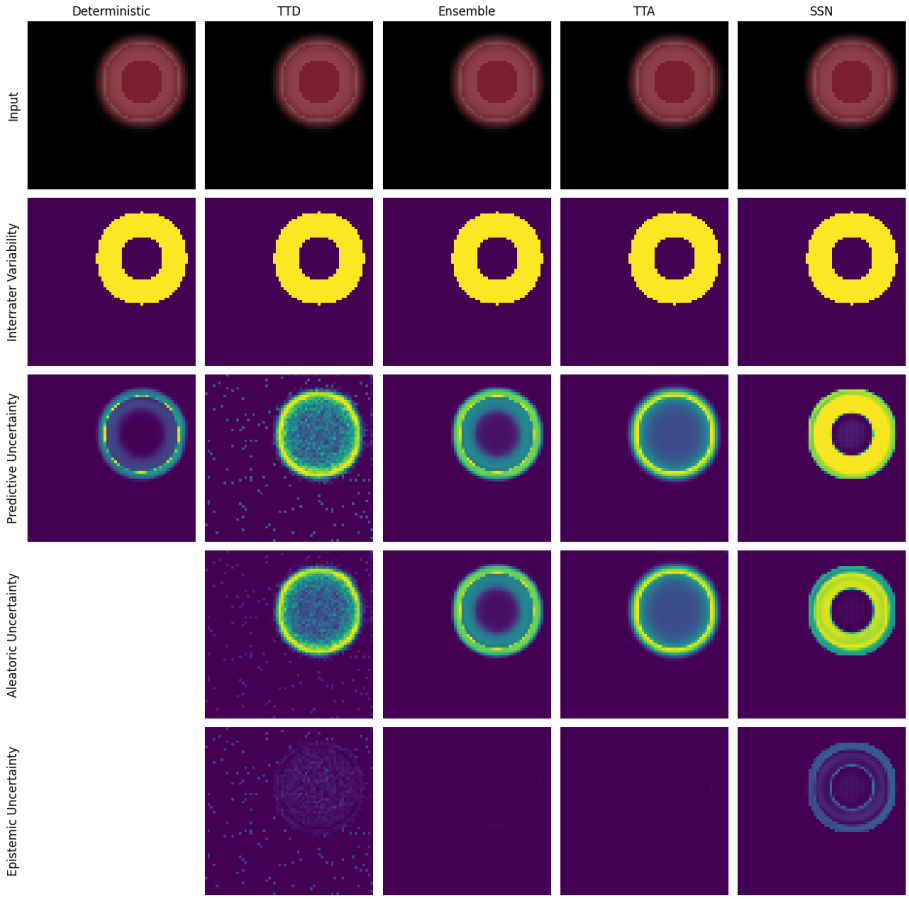


Figure B.6: Qualitative results for separating aleatoric and epistemic uncertainty for the toy dataset. The reference segmentations are shown as overlay over the input image. Further, the interrater variability based on the pixel variance is shown. The uncertainty scores per pixel are normalized between 0 and 0.5 for the deterministic model and between 0 and 0.7 for the other prediction models, reflecting the possible range of uncertainty values.

Qualitative results for the LIDC-IDRI datasets

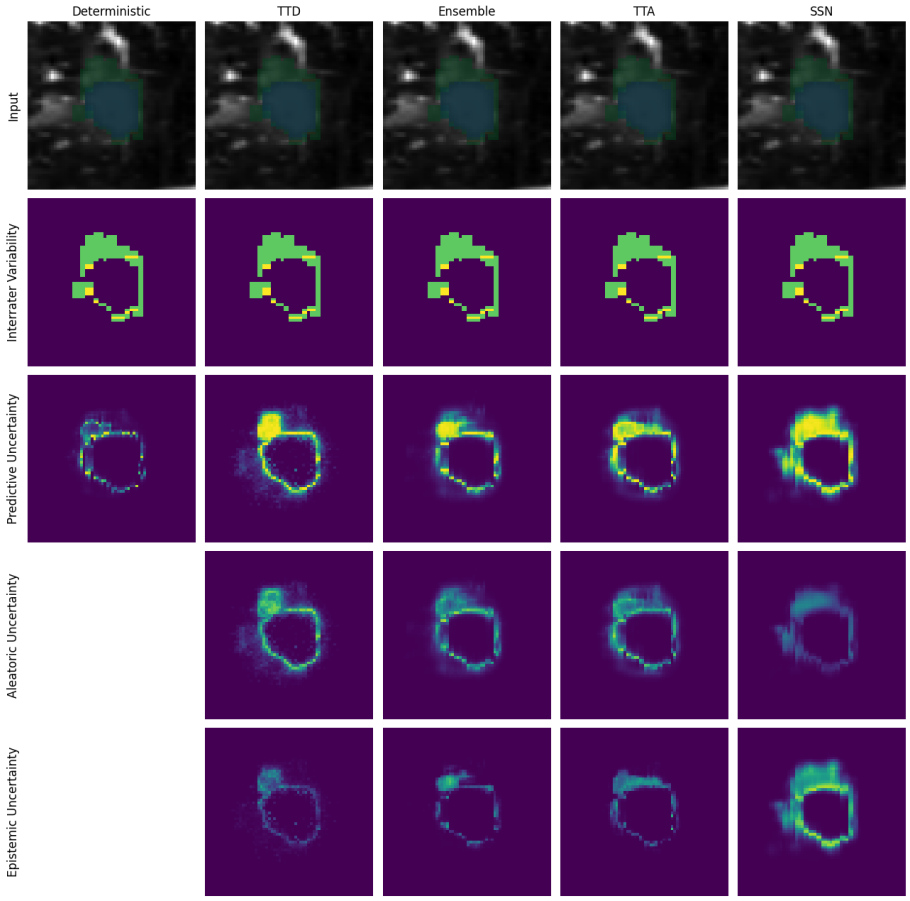


Figure B.7: Qualitative results for separating aleatoric and epistemic uncertainty for the LIDC TEX dataset. A case that is part of the i.i.d. test set is shown. The reference segmentations are shown as overlay over the input image. Further, the interrater variability based on the pixel variance is shown. The uncertainty scores per pixel are normalized between 0 and 0.5 for the deterministic model and between 0 and 0.7 for the other prediction models, reflecting the possible range of uncertainty values.

Texture shift i.i.d. example

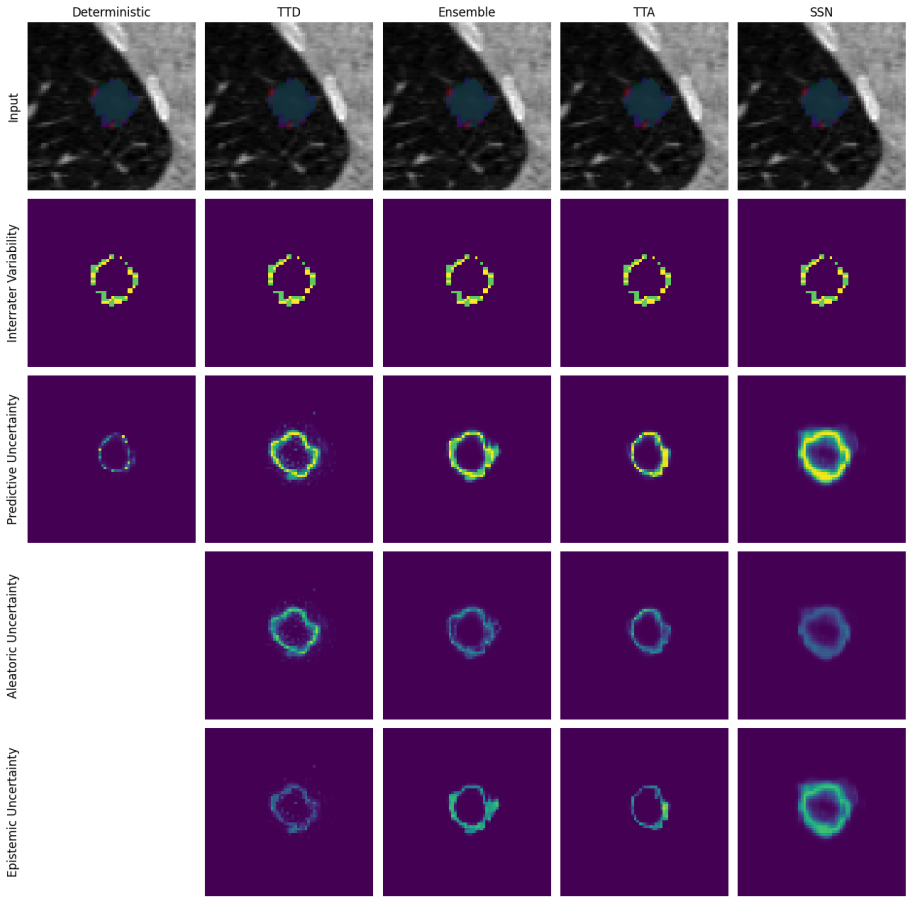


Figure B.8: Qualitative results for separating aleatoric and epistemic uncertainty for the LIDC TEX dataset. A case that is part of the OoD test set is shown. The reference segmentations are shown as overlay over the input image. Further, the interrater variability based on the pixel variance is shown. The uncertainty scores per pixel are normalized between 0 and 0.5 for the deterministic model and between 0 and 0.7 for the other prediction models, reflecting the possible range of uncertainty values.

Texture shift OoD example

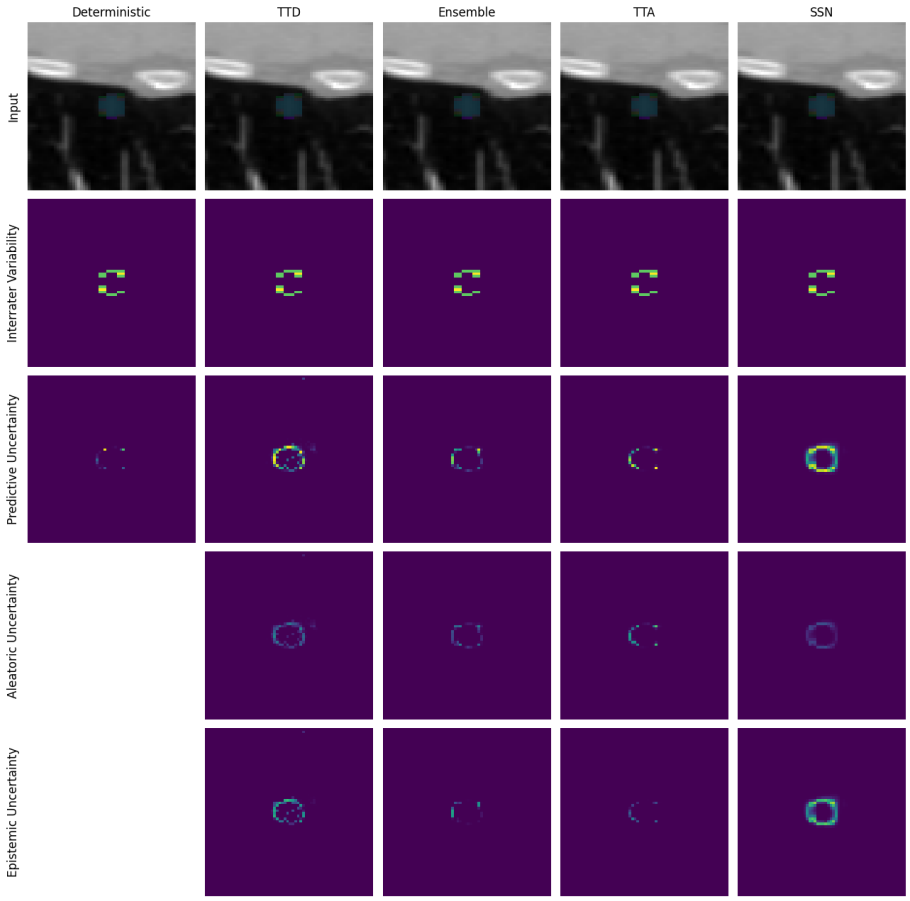


Figure B.9: Qualitative results for separating aleatoric and epistemic uncertainty for the LIDC MAL dataset. A case that is part of the i.i.d. test set is shown. The reference segmentations are shown as overlay over the input image. Further, the interrater variability based on the pixel variance is shown. The uncertainty scores per pixel are normalized between 0 and 0.5 for the deterministic model and between 0 and 0.7 for the other prediction models, reflecting the possible range of uncertainty values.

Malignancy shift i.i.d. example

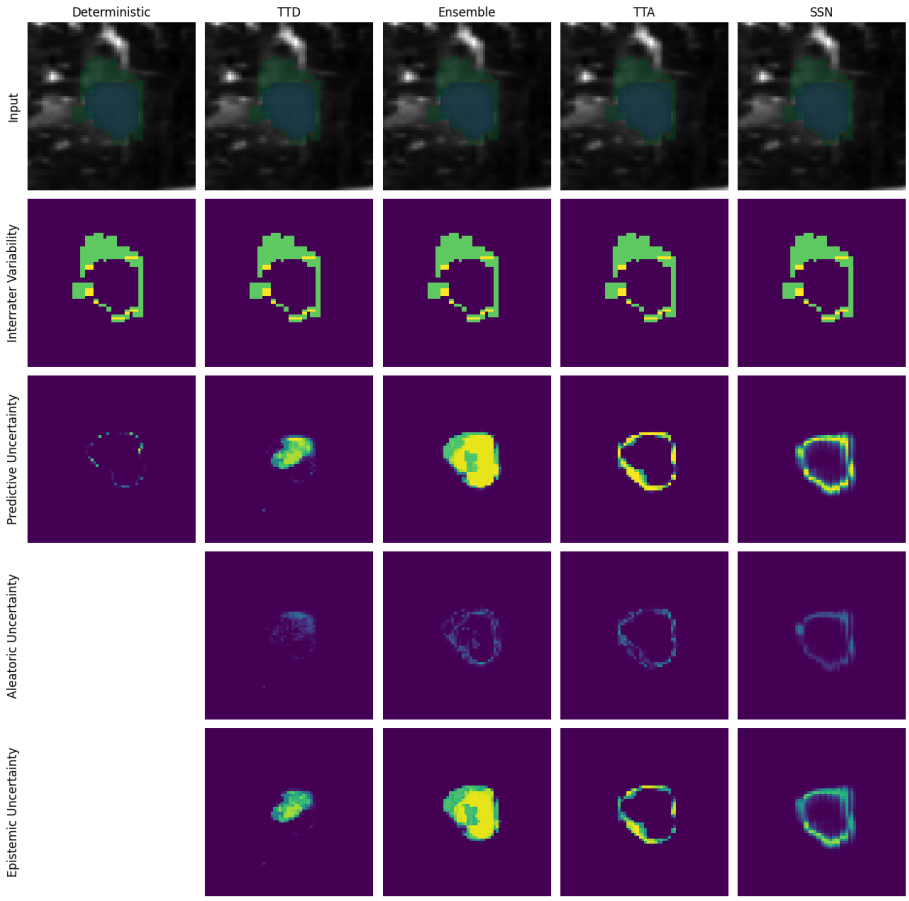


Figure B.10: Qualitative results for separating aleatoric and epistemic uncertainty for the LIDC MAL dataset. A case that is part of the OoD test set is shown. The reference segmentations are shown as overlay over the input image. Further, the interrater variability based on the pixel variance is shown. The uncertainty scores per pixel are normalized between 0 and 0.5 for the deterministic model and between 0 and 0.7 for the other prediction models, reflecting the possible range of uncertainty values.

Malignancy shift OoD example

Qualitative results for the GTA 5 / Cityscapes Dataset

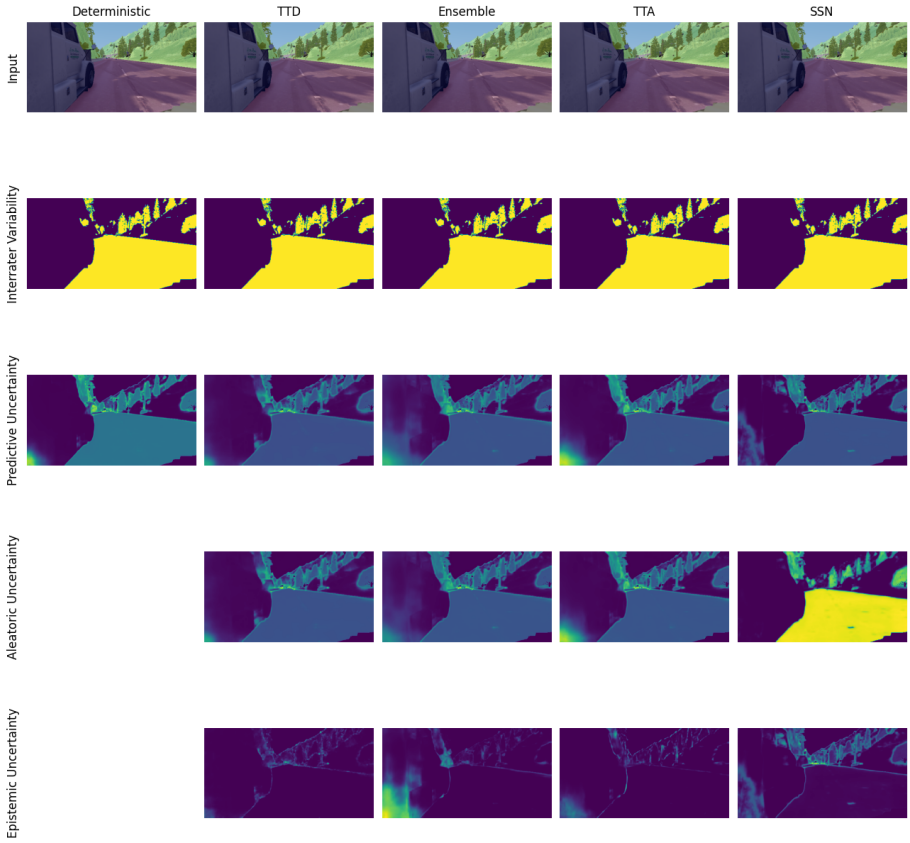


Figure B.11: Qualitative results for separating aleatoric and epistemic uncertainty for the GTA 5 / Cityscapes dataset. The reference segmentations are shown as overlay over the input image. Further, the interrater variability based on the pixel variance is shown. The uncertainty scores per pixel are normalized per image.

B.8 Detailed results of the evaluation on downstream tasks

The following tables show the detailed results on the downstream tasks as described in section 6.4.6. For the LIDC datasets, the results are shown in table B.5, while for the GTA5/CS dataset, the results are shown in table B.6.

Table B.5: **Evaluation of downstream tasks on the LIDC datasets.** The table shows the segmentation performance by means of the Dice score and evaluation metrics for 5 different downstream tasks, where ↑ depict higher scores are better and ↓ lower scores are better. All scores are multiplyed by 10^2. The color heatmap is normalized per column and per shift, brighter columns imply better scores. For AL, the second cycle was only executed with EU and PU, indicated by empty grey entries for AU. Reported results show the mean and standard deviation over 3 different seeds. Abbreviations: PM: Prediction model, UM: Uncertainty measure, UT: Modeled uncertainty Type (according to theory), AGG: Aggregation strategy.

					Seg. Performance		OoD-D	Failure Detection				AL	Calibration		Ambiguity Modeling			
Shift	PM	UM	UT	AGG	Dice i.i.d. ↑	Dice OoD ↑	AUROC ↑	AURC i.i.d. ↓	AURC OoD ↓	EAURC i.i.d. ↓	EAURC OoD ↓	AL Improv. OoD ↑	ACE i.i.d. ↓	ACE OoD ↓	NCC i.i.d. ↑	NCC OoD ↑	GED i.i.d. ↓	GED OoD ↓
Texture-Shift	Determ.	MSR	PU	Patch	81.42±0.28	64.05±0.1	46.43±2.28	19.18±0.24	41.27±1.19	6.13±0.09	18.64±1.27	-0.78±1.93	29.92±0.7	32.58±0.99	31.55±0.26	20.09±1.57	22.18±0.53	52.06±0.18
				Thresh	81.42±0.28	64.05±0.1	52.02±11.21	17.7±1.02	41.82±1.54	4.65±0.82	19.19±1.13	-2.57±3.49	29.92±0.7	32.58±0.99	31.55±0.26	20.09±1.57	22.18±0.53	52.06±0.18
	TTD	PE	PU	Patch	81.1±0.14	63.9±1.43	46.35±1.77	19.97±0.09	42.52±2.12	6.52±0.16	19.05±1.18	1.06±1.0	19.47±0.67	21.98±1.04	51.17±0.13	36.58±1.73	16.83±0.26	40.84±2.01
				Thresh	81.1±0.14	63.9±1.43	60.95±2.3	17.35±0.51	37.97±1.82	3.9±0.42	14.5±1.02	0.9±0.76	19.47±0.67	21.98±1.04	51.17±0.13	36.58±1.73	16.83±0.26	40.84±2.01
		EE	AU	Patch	81.1±0.14	63.9±1.43	43.41±1.38	20.25±0.04	41.65±3.65	6.8±0.13	18.19±2.78		26.95±0.27	26.86±1.66	51.74±0.26	37.21±2.13	16.83±0.26	40.84±2.01
				Thresh	81.1±0.14	63.9±1.43	40.4±1.68	20.03±0.25	42.94±2.22	6.57±0.17	18.77±1.58		26.95±0.27	26.86±1.66	51.74±0.26	37.21±2.13	16.83±0.26	40.84±2.01
		MI	EU	Patch	81.1±0.14	63.9±1.43	51.55±0.9	19.51±0.12	42.15±1.92	6.06±0.09	18.69±1.01	2.05±2.57	25.42±0.12	27.82±1.35	46.45±0.35	33.33±1.41	16.83±0.26	40.84±2.01
				Thresh	81.1±0.14	63.9±1.43	65.2±1.38	16.86±0.28	36.3±2.34	3.41±0.18	12.83±1.78	3.24±1.92	25.42±0.12	27.82±1.35	46.45±0.35	33.33±1.41	16.83±0.26	40.84±2.01
	Ensemble	PE	PU	Patch	82.34±0.18	64.36±0.85	54.52±2.55	15.24±0.25	37.06±4.43	5.6±0.2	15.05±3.67	1.02±3.04	19.96±0.91	24.7±0.54	47.82±0.55	34.62±1.03	16.05±0.35	40.37±1.09
				Thresh	82.34±0.18	64.36±0.85	66.47±5.2	16.07±0.39	38.38±2.84	3.59±0.37	16.37±2.06	0.56±2.53	19.96±0.91	24.7±0.54	47.82±0.55	34.62±1.03	16.05±0.35	40.37±1.09
		EE	AU	Patch	82.34±0.18	64.36±0.85	48.57±0.74	18.53±0.26	37.49±5.1	6.05±0.17	15.47±4.41		27.21±0.28	29.18±1.92	49.31±0.46	35.52±0.82	16.05±0.35	40.37±1.09
				Thresh	82.34±0.18	64.36±0.85	52.66±4.31	16.7±0.35	41.57±2.44	4.22±0.31	19.55±1.97		27.21±0.28	29.18±1.92	49.31±0.46	35.52±0.82	16.05±0.35	40.37±1.09
		MI	EU	Patch	82.34±0.18	64.36±0.85	65.4±3.71	17.55±0.29	38.15±5.64	5.06±0.22	16.13±4.92	0.96±3.48	28.71±0.38	31.35±1.58	39.05±0.8	30.1±0.84	16.05±0.35	40.37±1.09
				Thresh	82.34±0.18	64.36±0.85	72.1±5.05	15.58±0.32	36.76±2.92	3.1±0.3	14.75±2.13	-0.75±1.84	28.71±0.38	31.35±1.58	39.05±0.8	30.1±0.84	16.05±0.35	40.37±1.09
	TTA	PE	PU	Patch	81.83±0.3	65.08±0.45	50.95±1.4	18.57±0.25	36.28±1.96	5.76±0.12	14.38±1.92	0.45±1.97	22.78±1.01	26.07±2.37	46.04±0.31	31.51±0.22	17.37±0.6	41.49±0.9
				Thresh	81.83±0.3	65.08±0.45	59.8±10.85	16.77±0.52	39.21±0.91	3.97±0.38	17.31±0.78	-0.25±1.44	22.78±1.01	26.07±2.37	46.04±0.31	31.51±0.22	17.37±0.6	41.49±0.9
		EE	AU	Patch	81.83±0.3	65.08±0.45	49.2±1.89	18.06±0.09	39.39±0.5	6.15±0.13	17.49±0.62		26.45±0.15	28.32±1.87	47.57±0.47	33.48±0.33	17.37±0.6	41.49±0.9
				Thresh	81.83±0.3	65.08±0.45	48.93±11.85	17.6±0.77	40.74±1.09	4.79±0.65	18.84±0.96		26.45±0.15	28.32±1.87	47.57±0.47	33.48±0.33	17.37±0.6	41.49±0.9
		MI	EU	Patch	81.83±0.3	65.08±0.45	59.09±2.68	17.52±0.43	33.53±3.36	4.71±0.28	11.63±3.29	1.52±0.43	28.95±0.59	30.61±0.39	38.08±0.46	26.68±0.36	17.37±0.6	41.49±0.9
				Thresh	81.83±0.3	65.08±0.45	67.38±7.09	16.31±0.57	36.14±2.59	3.51±0.42	14.23±2.47	0.3±2.32	28.95±0.59	30.61±0.39	38.08±0.46	26.68±0.36	17.37±0.6	41.49±0.9
	SSN	PE	PU	Patch	81.28±0.16	59.51±1.79	53.57±3.02	19.6±0.42	40.79±6.31	6.41±0.39	16.2±5.62	-5.11±8.76	9.68±0.35	17.87±2.1	63.49±0.41	51.22±0.11	13.57±0.3	35.69±2.43
				Thresh	81.28±0.16	59.51±1.79	50.52±5.24	18.47±0.04	49.89±4.56	5.28±0.02	25.3±3.72	-7.24±8.79	9.68±0.35	17.87±2.1	63.49±0.41	51.22±0.11	13.57±0.3	35.69±2.43
		MI	AU	Patch	81.28±0.16	59.51±1.79	54.48±6.47	19.31±0.76	37.86±2.83	6.11±0.73	13.26±2.12		27.71±0.29	27.98±1.09	59.2±0.63	47.33±0.54	13.57±0.3	35.69±2.43
				Thresh	81.28±0.16	59.51±1.79	49.4±6.31	18.49±0.6	36.24±3.38	5.3±0.56	11.65±2.09		27.71±0.29	27.98±1.09	59.2±0.63	47.33±0.54	13.57±0.3	35.69±2.43
		EE	EU	Patch	81.28±0.16	59.51±1.79	58.13±3.43	19.54±0.33	41.57±6.22	6.34±0.3	16.98±5.69	-0.73±9.19	17.16±0.31	22.56±0.4	63.61±0.47	51.8±0.23	13.57±0.3	35.69±2.43
				Thresh	81.28±0.16	59.51±1.79	50.44±6.67	18.61±0.28	50.56±3.84	5.42±0.33	25.97±3.06	-5.7±10.94	17.16±0.31	22.56±0.4	63.61±0.47	51.8±0.23	13.57±0.3	35.69±2.43
Malignancy-Shift	Determ.	MSR	PU	Patch	78.97±0.29	65.03±3.36	86.44±2.33	19.79±0.16	36.1±9.33	4.77±0.27	17.87±8.27	0.79±8.0	32.17±0.96	28.41±1.45	28.39±1.24	20.15±1.78	26.42±0.63	54.57±5.45
				Thresh	78.97±0.29	65.03±3.36	58.94±4.42	21.86±0.19	48.62±8.06	6.84±0.38	30.39±6.94	-1.65±7.86	32.17±0.96	28.41±1.45	28.39±1.24	20.15±1.78	26.42±0.63	54.57±5.45
	TTD	PE	PU	Patch	78.6±0.41	67.84±1.62	90.13±0.07	20.95±0.51	22.93±3.61	4.85±0.34	5.39±4.15	2.27±4.26	21.09±0.63	19.13±0.82	47.88±0.27	35.6±1.53	19.88±0.74	39.09±2.89
				Thresh	78.6±0.41	67.84±1.62	73.08±7.04	22.08±0.69	25.97±3.92	6.33±0.07	8.43±4.47	2.68±4.48	21.09±0.63	19.13±0.82	47.88±0.27	35.6±1.53	19.88±0.74	39.09±2.89
		EE	AU	Patch	78.6±0.41	67.84±1.62	90.5±0.56	20.95±0.38	27.35±5.72	4.81±0.43	9.81±6.14		28.34±0.17	24.24±0.63	48.22±0.48	38.51±1.22	19.88±0.74	39.09±2.89
				Thresh	78.6±0.41	67.84±1.62	87.62±1.42	22.29±0.75	30.51±1.95	6.54±0.71	21.07±2.45		28.34±0.17	24.24±0.63	48.22±0.48	38.51±1.22	19.88±0.74	39.09±2.89
		MI	EU	Patch	78.6±0.41	67.84±1.62	88.12±1.29	20.65±0.34	21.96±2.13	4.85±0.36	4.43±2.68	1.84±4.39	26.6±0.38	24.43±0.37	44.55±0.08	31.08±0.89	19.88±0.74	39.09±2.89
				Thresh	78.6±0.41	67.84±1.62	90.01±1.24	21.62±0.58	24.2±2.91	6.37±0.63	6.66±3.4	1.55±4.4	26.6±0.38	24.43±0.37	44.55±0.08	31.08±0.89	19.88±0.74	39.09±2.89
	Ensemble	PE	PU	Patch	79.36±0.13	65.34±0.65	91.31±0.14	19.12±0.1	24.77±2.49	4.41±0.06	6.98±2.71	2.15±2.77	22.93±0.44	21.13±0.25	42.85±0.15	32.62±0.58	20.51±0.28	36.79±1.42
				Thresh	79.36±0.13	65.34±0.65	72.07±7.09	19.75±0.17	28.75±2.94	5.04±0.18	10.97±3.16	0.85±2.98	22.93±0.44	21.13±0.25	42.85±0.15	32.62±0.58	20.51±0.28	36.79±1.42
		EE	AU	Patch	79.36±0.13	65.34±0.65	89.39±0.45	19.56±0.14	24.07±2.8	4.85±0.11	10.28±2.33		29.48±0.47	26.4±0.83	44.23±0.39	35.38±0.73	20.51±0.28	36.79±1.42
				Thresh	79.36±0.13	65.34±0.65	53.44±2.59	22.37±0.29	52.64±1.34	7.66±0.32	34.85±1.41		29.48±0.47	26.4±0.83	44.23±0.39	35.38±0.73	20.51±0.28	36.79±1.42
		MI	EU	Patch	79.36±0.13	65.34±0.65	89.42±0.73	18.75±0.01	24.43±1.94	4.04±0.04	6.65±2.17	1.94±2.62	30.72±0.07	27.43±0.31	36.3±0.17	26.97±0.66	20.51±0.28	36.79±1.42
				Thresh	79.36±0.13	65.34±0.65	75.02±0.96	18.95±0.36	26.68±2.47	4.24±0.39	8.9±2.67	0.92±2.93	30.72±0.07	27.43±0.31	36.3±0.17	26.97±0.66	20.51±0.28	36.79±1.42
	TTA	PE	PU	Patch	79.08±0.12	65.2±3.68	88.41±1.02	19.24±0.14	28.94±7.15	4.39±0.23	11.07±6.19	0.17±8.74	25.13±0.34	21.54±0.25	40.89±0.61	30.39±2.45	21.85±0.26	42.45±4.5
				Thresh	79.08±0.12	65.2±3.68	66.79±2.58	20.34±0.74	35.83±7.94	5.5±0.42	17.97±7.13	0.92±8.45	25.13±0.34	21.54±0.25	40.89±0.61	30.39±2.45	21.85±0.26	42.45±4.5
		EE	AU	Patch	79.08±0.12	65.2±3.68	86.63±1.94	19.64±0.11	35.61±9.85	4.8±0.39	17.75±8.61		29.23±0.79	25.37±0.71	42.46±0.47	32.91±2.11	21.85±0.26	42.45±4.5
				Thresh	79.08±0.12	65.2±3.68	52.66±5.18	22.31±0.31	51.72±8.34	7.46±0.23	33.86±7.08		29.23±0.79	25.37±0.71	42.46±0.47	32.91±2.11	21.85±0.26	42.45±4.5
		MI	EU	Patch	79.08±0.12	65.2±3.68	86.32±2.14	18.75±0.19	28.06±7.07	3.94±0.32	10.19±6.25	0.87±8.57	30.86±0.3	26.51±0.73	34.99±1.08	25.15±2.15	21.85±0.26	42.45±4.5
				Thresh	79.08±0.12	65.2±3.68	70.06±3.16	19.26±0.76	31.25±7.73	4.42±0.38	13.39±6.89	0.49±0.23	30.86±0.3	26.51±0.73	34.99±1.08	25.15±2.15	21.85±0.26	42.45±4.5
	SSN	PE	PU	Patch	79.05±0.24	67.79±1.75	83.59±0.6	20.42±0.68	23.09±1.64	5.57±0.44	5.7±1.4	1.53±5.39	11.62±0.24	13.59±0.32	61.14±1.11	47.48±0.03	16.79±0.3	36.2±2.02
				Thresh	79.05±0.24	67.79±1.75	71.97±1.26	20.74±0.13	24.03±1.41	5.83±0.26	6.01±1.93	0.62±4.99	11.62±0.24	13.59±0.32	61.14±1.11	47.48±0.03	16.79±0.3	36.2±2.02
		MI	AU	Patch	79.05±0.24	67.79±1.75	71.68±3.1	20.23±0.98	26.71±2.76	5.38±0.79	9.32±2.43		30.31±0.07	25.93±0.37	55.42±1.71	44.16±0.26	16.79±0.3	36.2±2.02
				Thresh	79.05±0.24	67.79±1.75	56.56±8.8	20.63±1.3	31.75±3.27	5.77±1.15	14.36±3.14		30.31±0.07	25.93±0.37	55.42±1.71	44.16±0.26	16.79±0.3	36.2±2.02
		EE	EU	Patch	79.05±0.24	67.79±1.75	85.3±1.15	20.48±0.49	23.74±1.79	5.62±0.28	6.35±1.63	1.23±5.06	17.67±0.62	19.66±0.28	61.82±0.68	47.3±0.08	16.79±0.3	36.2±2.02
				Thresh	79.05±0.24	67.79±1.75	67.74±4.7	20.56±0.41	27.02±3.95	5.71±0.69	20.53±5.53	0.38±5.55	17.67±0.62	19.66±0.28	61.82±0.68	47.3±0.08	16.79±0.3	36.2±2.02

Table B.6: **Evaluation of downstream tasks on the GTA 5 / Cityscapes dataset.** The table shows the segmentation performance by means of the Dice score and evaluation metrics for 5 different downstream tasks, where ↑ depict higher scores are better and ↓ lower scores are better. All scores are multiplyed by 10^2. The color heatmap is normalized per column, brighter columns imply better scores. For AL, the second cycle was only executed with EU and PU, indicated by empty grey entries for AU. Reported results show the mean and standard deviation over 3 different seeds. Abbreviations: PM: Prediction model, UM: Uncertainty measure, UT: Modeled uncertainty Type (according to theory), AGG: Aggregation strategy.

				Seg. Performance		OoD-D	Failure Detection				AL	Calibration		Ambiguity Modeling			
PM	UM	UT	AGG	Dice i.i.d. ↑	Dice OoD ↑	AUROC ↑	AURC i.i.d. ↓	AURC OoD ↓	EAURC i.i.d. ↓	EAURC OoD ↓	AL Improv. OoD ↑	ACE i.i.d. ↓	ACE OoD ↓	NCC i.i.d. ↑	NCC OoD ↑	GED i.i.d. ↓	GED OoD ↓
Determ.	MSR	PU	Patch	71.73±0.13	58.37±0.51	33.43±1.16	26.52±0.23	39.24±0.75	7.28±0.15	8.16±0.72	-0.5±1.78	13.09±0.15	17.21±0.31	50.61±0.43	46.81±0.61	36.09±0.26	58.87±0.83
			Image	71.73±0.13	58.37±0.51	69.99±0.58	24.2±0.17	36.85±0.33	4.96±0.04	5.76±0.24	1.44±1.73	13.09±0.15	17.21±0.31	50.61±0.43	46.81±0.61	36.09±0.26	58.87±0.83
TTD	PE	PU	Patch	71.63±0.14	58.62±0.91	36.67±1.42	26.77±0.2	39.46±0.57	7.43±0.06	8.43±0.35	-0.29±2.72	15.14±0.11	15.07±0.07	26.68±0.29	25.57±3.5	34.03±0.28	56.08±1.61
			Image	71.63±0.14	58.62±0.91	67.8±0.77	24.51±0.21	36.14±0.3	5.16±0.06	5.12±0.09	0.94±2.81	15.14±0.11	15.07±0.07	26.68±0.29	25.57±3.5	34.03±0.28	56.08±1.61
	EE	AU	Patch	71.63±0.14	58.62±0.91	36.83±1.35	26.76±0.21	39.45±0.61	7.41±0.07	8.43±0.38		15.08±0.11	15.18±0.08	27.67±0.29	26.39±3.46	34.03±0.28	56.08±1.61
			Image	71.63±0.14	58.62±0.91	68.03±0.75	24.5±0.21	36.15±0.3	5.15±0.06	5.13±0.09		15.08±0.11	15.18±0.08	27.67±0.29	26.39±3.46	34.03±0.28	56.08±1.61
	MI	EU	Patch	71.63±0.14	58.62±0.91	40.35±4.04	26.88±0.11	39.18±0.6	7.24±0.05	8.15±0.39	0.03±2.86	17.56±0.13	17.82±0.5	-23.17±0.19	-12.85±2.61	34.03±0.28	56.08±1.61
			Image	71.63±0.14	58.62±0.91	51.08±2.53	26.05±0.24	36.61±0.38	6.71±0.09	5.58±0.16	0.45±2.82	17.56±0.13	17.82±0.5	-23.17±0.19	-12.85±2.61	34.03±0.28	56.08±1.61
Ensemble	PE	PU	Patch	71.92±0.14	59.36±0.46	32.72±0.57	26.4±0.13	38.33±0.63	7.32±0.07	7.98±0.55	-0.65±1.67	15.64±0.12	15.61±0.22	23.77±0.11	25.37±0.98	31.92±0.22	50.12±0.36
			Image	71.92±0.14	59.36±0.46	72.06±0.8	24.02±0.12	35.51±0.83	5.05±0.01	5.16±0.17	1.17±1.73	15.64±0.12	15.61±0.22	23.77±0.11	25.37±0.98	31.92±0.22	50.12±0.36
	EE	AU	Patch	71.92±0.14	59.36±0.46	28.69±0.35	26.5±0.15	38.79±0.42	7.63±0.05	8.4±0.41		15.57±0.11	15.31±0.21	26.74±0.14	29.59±0.9	31.92±0.22	50.12±0.36
			Image	71.92±0.14	59.36±0.46	66.81±0.94	24.05±0.11	35.54±0.78	5.08±0.01	5.19±0.22		15.57±0.11	15.31±0.21	26.74±0.14	29.59±0.9	31.92±0.22	50.12±0.36
	MI	EU	Patch	71.92±0.14	59.36±0.46	90.63±0.29	26.69±0.04	38.57±1.15	7.72±0.13	8.22±0.61	1.77±1.79	21.33±0.06	17.43±0.28	-22.55±0.08	-6.06±0.79	31.92±0.22	50.12±0.36
			Image	71.92±0.14	59.36±0.46	90.05±0.29	25.68±0.09	36.51±1.29	6.71±0.08	6.16±0.63	1.85±1.8	21.33±0.06	17.43±0.28	-22.55±0.08	-6.06±0.79	31.92±0.22	50.12±0.36
TTA	PE	PU	Patch	71.82±0.13	58.39±0.47	31.66±0.71	26.54±0.26	39.44±0.94	7.38±0.15	8.39±1.07	-0.61±1.71	15.37±0.14	15.42±0.31	24.84±0.23	28.14±1.28	33.52±0.26	54.06±0.72
			Image	71.82±0.13	58.39±0.47	69.81±0.32	24.27±0.18	36.44±0.16	5.11±0.05	5.39±0.05	1.44±1.78	15.37±0.14	15.42±0.31	24.84±0.23	28.14±1.28	33.52±0.26	54.06±0.72
	EE	AU	Patch	71.82±0.13	58.39±0.47	28.85±0.8	26.55±0.27	39.39±0.98	7.39±0.17	8.35±1.12		15.3±0.14	15.32±0.33	26.29±0.24	30.47±1.19	33.52±0.26	54.06±0.72
			Image	71.82±0.13	58.39±0.47	66.61±0.31	24.28±0.18	36.42±0.16	5.12±0.05	5.37±0.05		15.3±0.14	15.32±0.33	26.29±0.24	30.47±1.19	33.52±0.26	54.06±0.72
	MI	EU	Patch	71.82±0.13	58.39±0.47	92.75±0.11	27.1±0.09	39.65±0.47	7.67±0.05	8.01±0.41	2.12±1.77	22.04±0.39	19.1±0.32	-16.23±0.14	-4.45±1.2	33.52±0.26	54.06±0.72
			Image	71.82±0.13	58.39±0.47	94.32±0.33	26.12±0.05	37.57±0.41	6.96±0.08	6.53±0.54	2.46±1.77	22.04±0.39	19.1±0.32	-16.25±0.14	-4.45±1.2	33.52±0.26	54.06±0.72
SSN	PE	PU	Patch	65.49±0.13	51.9±0.44	77.95±1.28	33.85±0.35	46.53±0.89	10.14±0.29	9.91±0.55	2.01±2.21	16.1±0.13	17.84±0.41	55.97±0.09	37.26±0.95	25.6±0.03	45.26±1.1
			Image	65.49±0.13	51.9±0.44	82.0±1.1	30.07±0.05	43.05±0.95	6.36±0.14	6.43±0.41	3.21±2.25	16.1±0.13	17.84±0.41	55.97±0.09	37.26±0.95	25.6±0.03	45.26±1.1
	AU	AU	Patch	65.49±0.13	51.9±0.44	53.1±3.37	32.57±0.31	46.28±1.29	8.85±0.4	9.65±1.0		21.0±0.14	25.8±0.42	70.34±0.36	51.8±0.44	25.6±0.03	45.26±1.1
			Image	65.49±0.13	51.9±0.44	55.21±1.37	31.0±0.09	46.07±0.67	7.29±0.13	9.44±0.51		21.6±0.14	25.8±0.42	70.34±0.36	51.8±0.44	25.6±0.03	45.26±1.1
	EE	EU	Patch	65.49±0.13	51.9±0.44	78.12±1.74	33.87±0.42	46.86±0.86	10.16±0.37	10.23±0.39	2.44±2.17	10.49±0.31	11.12±0.45	4.57±1.07	2.95±0.9	25.6±0.03	45.26±1.1
			Image	65.49±0.13	51.9±0.44	86.2±1.08	32.58±0.43	43.62±1.1	8.87±0.41	7.0±0.56	2.37±1.91	10.49±0.31	11.12±0.45	4.57±1.07	2.95±0.9	25.6±0.03	45.26±1.1

Appendix C

Principled Evaluation of Active Learning for 3D Biomedical Segmentation

This chapter of the Appendix uses the Appendix of Carsten T. Lüth, Jeremias Traub, Kim-Celine Kahl, Till J. Bungert, Lukas Klein, Lars Krämer, Paul F. Jaeger, Fabian Isensee, and Klaus Maier-Hein (2025). "nnActive: A Framework for Evaluation of Active Learning in 3D Biomedical Segmentation". In: *Transactions on Machine Learning Research*.

C.1 Related Works

Pitfalls

We present a detailed comparison of related works and their evaluation protocols in table C.1.

The scoring rules for our pitfalls criteria in table 7.1 used for ✔, (✔) and a ✗ if not addressed:

P1 A. Evaluate performance on at least 3 datasets (only counting 3D biomedical). (✔)
B. Evaluate at least two different starting budgets and query sizes. (✔)
If A. & B. ✔

P2 Use 3D models with training optimized for partial annotations. ✔
2D models: pretrained, Semi-Supervised Training or partial annotations. (✔)

P3 Evaluate Random Baselines that take into account that for 3D Biomedical image datasets large areas of the images are pure background and/or make use of the 3D structure of the data. ✔

P4 Use metrics that take into account that the effort to annotate background is very low compared to foreground. ✔

Compared Literature

Nath et al. (2021) Contribution: Propose to query samples from dataset pool without removing them and enforce diversity with Mutual Information over histogramms.
Query Methods: BALD, BALD with Mutual Information on Histogramms, Random.
Datasets: MSD Hippocampus and Pancreas
Evaluation Metric: Best Mean Dice (3D) over Experiment
Evaluated Query Sizes per dataset (max): 1
Evaluated Starting Budgets per dataset(max): 1

P1 3 biomedical datasets, no ablations for multiple annotation budgets.

P2 Do use 3D models but no partial annotations, also no pretrained models or 3D models in combination with partial annotations and also no Data Augmentations.

P3 No improved random baseline.

P4 No Measurement taking into account the annotation effort.

Burmeister et al., 2022 Contribution: Evaluate Strided and Stratified Sampling Strategies.
Query Methods: Least Confidence, Entropy, Distance-based representativeness sampling, Cluster-based representativeness sampling, Strided Random Sampling and Stratified Random Sampling. Additional experiments with label interpolation.
Datasets: MSD Hippocampus, Prostate and Heart
Evaluation Metric: Mean Dice (3D) Plots.
Evaluated Query Sizes per dataset (max): 1
Evaluated Starting Budgets per dataset(max): 1

P1 3 biomedical datasets, not multiple annotation budgets per dataset.
P2 Does neither use pretrained models or 3D models in combination with partial annotations.
P3 Do use Strided and Stratified Random Sampling.
P4 No Measurement taking into account the annotation effort.

Gaillochet et al. (2023a) Contribution: Propose Stochastic Batches as Query Methods.
Query Methods: Stochastic Batches, Entropy, BALD, Test-Time Augmentations, Learning Loss, Core-Set, Random.
Datasets: Prostate MR Image Segmentation (PROMISE) challenge 2012, MSD Hippocampus
Evaluation Metric: Mean Dice (3D) and Hausdorff Distance.
Evaluated Query Sizes per dataset (max): 3 (ablation one dataset)
Evaluated Starting Budgets per dataset(max): 3 (ablation one dataset)

P1 2 biomedical datasets, not multiple annotation budgets per dataset; however, multiple annotation budget ablations for one dataset.
P2 Does neither use pretrained models or 3D models in combination with partial annotations.
P3 No Improved Random Baselines.
P4 No Measurement taking into account the annotation effort.

Gaillochet et al., 2023b Contribution: Propose Test-Time Augmentations as Query Method.
Query Methods: Entropy, BALD, Test-Time Augmentations, Core-Set, Random.
Datasets: ACDC
Evaluation Metric: Mean Dice (2D and 3D).
Evaluated Query Sizes per dataset (max): 1
Evaluated Starting Budgets per dataset(max): 1

P1 1 biomedical datasets, not multiple annotation budgets per dataset, however, multiple annotation budget ablations for one dataset.
P2 Use 2D Semi-Supervised models.
P3 No Improved Random Baselines.
P4 No Measurement taking into account the annotation effort.

S. Ma et al. (2024) Contribution: Add target & boundary awareness to existing Query Methods.
Query Methods: Entropy (with and without Dropout), BALD, Margin Sampling, Least Confidence.
Datasets: MSD Spleen, BraTS
Evaluation Metric: Mean Dice (2D and 3D) – % required data to achieve fully annotated performance and peak performance.
Evaluated Query Sizes per dataset (max): 1
Evaluated Starting Budgets per dataset(max): 1

P1 1 biomedical datasets, not multiple annotation budgets per dataset, however, multiple annotation budget ablations for one dataset.
P2 Does neither use pretrained models or 3D models in combination with partial annotations.
P3 No Improved Random Baselines.
P4 No Measurement taking into account the annotation effort.

Föllmer et al., 2024 Contribution: Propose Uncertainty-Aware Subomdular Information Measure (USIM) as Query Method.
Query Methods: USIMF, USIMC, Mean STD, Core-Set, BADGE (LL), Stochastic Batches, Entropy,

BALD, Random.
Datasets: MSD Spleen, Liver and Hippocampus
Evaluation Metric: Mean Dice (3D) – Pairwise Penalty Matrix.
Evaluated Query Sizes per dataset (max): 1
Evaluated Starting Budgets per dataset(max): 1

P1 3 biomedical datasets, not multiple annotation budgets per dataset, however, multiple annotation budget ablations for one dataset.
P2 Use 2D Semi-Supervised models.
P3 No Improved Random Baselines.
P4 No Measurement taking into account the annotation effort.

Vepa et al. (2024) Contribution: Propose Metric Learning Based Query Method building upon Core-Set (Core-Metric).
Query Methods: Core-Metric, Core-Set, Random, CoreGCN, TypiClust, Stochastic Batches, VAAL, Variance Ratio, BALD.
Datasets: ACDC, CHAOS (Combined Healthy Abdominal Organ Segmentation), MS-CMR (Multi-sequence Cardiac MR Segmentation Challenge) and DAVIS (Densely Annotated Video Segmentation)[1]
Evaluation Metric: Mean Dice (3D) – Pairwise Penalty Matrix.
Evaluated Query Sizes per dataset (max): 2 (1 Pretrained and 1 Trained from Scratch)
Evaluated Starting Budgets per dataset(max): 2 (1 Pretrained and 1 Trained from Scratch)

P1 3 biomedical datasets and multiple annotation budget ablations for one dataset.
P2 Use 2D models, both pretrained and trained from random initialization.
P3 No Improved Random Baselines.
P4 No Measurement taking into account the annotation effort.

J. Shi et al. (2024) Contribution: Propose Predictive Accuracy-based Active Learning (PAAL).
Query Methods: Random, Entropy, Variation Ratio, Margin, KMeans, CoreSet, Entropy+KMeans, AB-UNet, CEAL, LPL, PAAL
Datasets: ACDC, SegThor, MSD Brain, Liver OAR (in-house dataset)
Peculiarity: Use 1/5th of the data as a validation set used during training to determine whether the query step will be performed.
Evaluation Metric: Mean Dice (not specified whether 2D or 3D in paper and not clearly described in code)
Evaluated Query Sizes per dataset (max): 3
Evaluated Starting Budgets per dataset (max): 3

P1 4 biomedical datasets, no multiple annotation budgets per dataset.
P2 Does neither use pretrained models or 3D models in combination with partial annotations.
P3 No Improved Random Baselines.
P4 Show the number of slices for all classes for different QMs.

Ours Query Methods: BALD, Entropy, PowerBALD, SoftrankBALD, PowerPE, Random, Random 66%FG, Random 33%FG. Datasets: ACDC, AMOS, Hippocampus, KiTS Evaluation Metrics: Mean Dice (3D) – Pairwise Penalty Matrix, Area Under Budget Curve, Final Mean Dice, Foreground Efficiency.
Evaluated Query Sizes per dataset (max): 3
Evaluated Starting Budgets per dataset (max): 3

P1 4 biomedical datasets with experiments on three different Label Regimes each with one query size and starting budget.
P2 Using 3D models that are trained with a partial loss on the annotated regions.
P3 Random 33%FG and Random 66% FG alleviate background selection issue of Random.
P4 We propose the dedicated measure named *Foreground Efficiency* (FG-Eff) (see section C.4 for details).

[1]Not included in dataset count as it is a non-medical non-3D dataset

Table C.1: Comparison of works in the field of Active Learning for 3D biomedical imaging. **Notation**: #Datasets[‡]: Only counting 3D biomedical datasets; no[†]: not specified in paper and not found in code; **N.S.**: not specified in paper and code.

Name	Seg. Model	Full Retraining	Query Design	#Datasets[‡]	Novel QM	Ensemble	Seeds	Advanced Random	Training Strategies	Data Augmentations
Föllmer et al. (2024)	2D nnU-Net (v.1)	no	2D Slice	3	yes	no	2	no	Standard Training	yes
S. Ma et al. (2024)	2D U-Net	no	2D Slice	2	yes	no	N.S.	no	Standard Training	yes
Burmeister et al. (2022)	2D U-Net	no	2D Slice	3	no	no	3	yes	Standard Training	no[†]
Gaillochet et al. (2023b)	2D U-Net	yes	2D Slice	1	yes	no	5	no	Semi-SL	yes
Gaillochet et al. (2023a)	2D U-Net	yes	2D Slice	2	yes	no	5	no	Standard Training	yes
Nath et al. (2021)	3D U-Net	yes	3D Image	2	yes	yes	5	no	Standard Training	no
Vepa et al. (2024)	2D U-Net	yes	2D Slice	3	yes	no	5	no	Pre-Trained& 2D Partial Loss	yes
J. Shi et al. (2024)	2D U-Net	no	2D Slice	4	yes	no	5	no	Standard Training	yes
Ours	3D nnU-Net (v.2)	yes	3D Patch	4	no	yes	4	yes	Partial Loss	yes

C.2 Task Description

In Active Learning (AL) for 3D biomedical image segmentation, acquiring full annotations for an entire volumetric scan is often infeasible due to the extensive time required. Instead, partial annotations allow for selective labeling of subregions within a 3D image, reducing annotation effort while still guiding model learning effectively. This subsection formalizes the task of querying and incorporating partial annotations in a 3D AL framework.

Mathematical Formulation Let $\mathcal{X}$ denote the space of 3D volumetric images, where each sample is a 3D image $X \in \mathbb{R}^{M \times H \times W \times D}$, with number of modalities M, height H , width W , and depth D. The corresponding dense ground-truth segmentation is given by $Y \in \{0, 1, \ldots, C\}^{H \times W \times D}$, where C is the number of classes.

In a standard supervised learning setting, a model f_θ is trained using full annotations (X, Y) from a dataset $\mathcal{D} = \{(X^{(i)}, Y^{(i)})\}_{i=1}^{N}$. However, in AL with partial annotations, we define a Query Method that can select multiple subsets of the volume of a single image $Q(X)$ spread over the entire dataset. For a single image, the annotated subset is denoted as:

$$\tilde{Y} = Q(X), \quad \tilde{Y} \subseteq Y$$

where $\tilde{Y}$ represents the annotated queries where only a fraction of the full annotation is provided.

The unobserved regions remain unannotated and are ignored or used for weakly supervised training.

In this work, we focus on 3D patches for partial annotation. Thus, a partial annotation for one image is defined as $\tilde{Y} = \{Y_{h:h_p, w:w_p, d:d_p} \mid (h, w, d) \in \mathcal{S}_P\}$, with (h_p, w_p, d_p) denoting the size of the 3D patch and $\mathcal{S}_P$ the set of patch locations. [2]

Given a dataset $\mathcal{D} = \{(X^{(i)}, \tilde{Y}^{(i)})\}_{i=1}^{N}$, where only $\tilde{Y}^{(i)}$ is available for training, the loss function is adapted to account for missing labels:

$$\mathcal{L}(\theta) = \sum_{i=1}^{N} \sum_{j \in \mathcal{S}^{(i)}} \ell(f_\theta(X_j^{(i)}), \tilde{Y}_j^{(i)})$$

where $\mathcal{S}^{(i)}$ denotes the queried (labeled) locations in image i .

[2] 2D Slices represent a subset of 3D patches, defined by e.g. $h_p = H, w_p = W, d_p = 1$.

C.3 Active Learning Framework

Algorithm 1 Active Learning Patch Selection

Input:
Set of images $\{X^{(i)}\}_{i=1}^{N}$, query size n, labeled set $\mathcal{L}$, Uncertainty function U, Aggregation function A, o allowed overlap **Output:** Final query set $\mathcal{Q}$

Initialize final query set $\mathcal{Q} \leftarrow \emptyset$
for each image $X^{(i)} \in \{X^{(i)}\}_{i=1}^{N}$ **do**
  $\mathcal{U} \leftarrow U(X^{(i)}, \mathcal{M})$ # compute uncertainty for image
  $\mathcal{U}_{\text{Agg}} \leftarrow A(\mathcal{U})$ # aggregate uncertainties to patch-level
  $\mathcal{Q}_{\text{Image}} \leftarrow \emptyset$ # initialize best patches for current image
  for q in sort$(\mathcal{U}_{\text{Agg}})$[::-1] **do** # sort in descending order according to uncertainty
    if overlap$(q, \mathcal{Q}_{\text{Image}} \cup \mathcal{L}) \leq o$ **then** # ensure that
      $\mathcal{Q}_{\text{Image}} \leftarrow \mathcal{Q}_{\text{Image}} \cup \{q\}$
    end if
  end for
  $\mathcal{Q} \leftarrow \mathcal{Q} \cup \mathcal{Q}_{\text{Image}}$
end for
$\mathcal{Q} \leftarrow$ sort$(\mathcal{Q})$[::-1] # sort in descending according to uncertainty
Return $\mathcal{Q}$

To ensure that nnActive can be used both for benchmarking and in production, we perform all perturbations of the images inside of the nnU-Net dataset structure. More specifically, inside the *nnUNet_raw* folder where we also store *loop_XXX.json* files, which store all relevant information of the queried patches. This allows to change the labels of all images directly in-place. Changes in the *nnUnet_raw* folder are transferred to the preprocessed dataset used for training using the standard *nnUNet_preprocessing* step.

For the query stage we build it on the patchwise inference of nnU-Net in a final stage after each image is predicted for all ensemble members. The algorithm used in our framework for a top-k uncertainty method (e.g., BALD or Predictive Entropy) is outlined in algorithm 1.

To enrich the spatial context available to the model, we enhanced the standard patch-based nnU-Net trainer through region sampling. Specifically, the final patch used for a forward pass still contains at least one labeled voxel (based on random or class-specific sampling), but the patch is not centered on the annotated voxel (as for the standard nnU-Net trainer). Instead, the annotated voxel is randomly located within the final patch, following a uniform distribution over the valid patch region. Since not all voxels in the input patch are necessarily annotated, nnActive supports training with partial losses, applying the loss only where labels are available. Importantly, the patch size used during the model's forward pass is always determined by the nnU-Net plans and configurations, which is fixed for each dataset. The query patch size used in the nnActive experiment configuration is not necessarily identical to the nnU-Net patch size.

C.4 Evaluation Metrics

In our evaluation, we performed an analysis based on all of the metrics described in this subsection.

In the analysis of our main study, we focused on all metrics, whereas in our ablations, we put special emphasis on the AUBC and Final Dice as they allow easier direct comparisons of values. This is also visualized in the overview figure fig. C.2.

Our newly proposed metric, FG-Eff, measuring the annotation efficiency by proxy of foreground voxels, is described in section C.4.

Final Dice

We use the Final Dice value after the annotation budget is exhausted for evaluation, as it allows for easy interpretation and puts a special emphasis on later stages of AL experiments.

AUBC

We compute the Area Under the Budget Curve (AUBC) for each dataset and Label Regime based on the Mean Dice to allow assessing the absolute performance each QM brings (see (Zhan, H. Liu, et al., 2021; Zhan, Q. Wang, Huang, Xiong, Dou, and Antoni B. Chan, 2022b) for more details). It aggregates the results of one Label Regime using the trapezoid method, and higher values indicate better performance under all budgets of the Label Regime.

Our normalization of the AUBC is set so that if all values on one Label Regime are equal to 0.8, the AUBC will return 0.8.

Pairwise Penalty Matrix

We employ the Pairwise Penalty Matrix (PPM) to assess whether one QM significantly outperforms others in terms of Mean Dice. This metric reflects how frequently a method yields statistically superior performance compared to another, based on a two-sided t-test with a significance level of $\alpha = 0.05$ (see (J. T. Ash et al., 2020) for further details) and whether the mean performance of method i is larger than that of method j and vice-versa. The PPM enables aggregation across multiple datasets and Label Regimes, though it does not account for absolute performance differences.

In the final matrix, we show values in % where each row i represents the fraction of settings where method i significantly outperforms other methods, whereas each column j shows the fraction of settings where another significantly outperforms method j.

Foreground Efficiency

Overview We measure the annotation efficiency by proxy of the amount of foreground annotation using the decay parameter γ, we term Foreground Efficiency (FG-Eff) for an exponential decay fitted to the performance gap to a model trained on the entire dataset and the number of foreground voxels. It allows for a simpler interpretation of plots like the following: As the number of foreground voxels represents a proxy for annotation effort, the FG-Eff does not replace other performance metrics s.a. AUBC, Pairwise Pen but should be seen as an extension of them.

Mathematical Definition The formula for the fitted exponential decay is given in eq. (C.1), where values with a $\hat{}$ are estimated empirically based on the data prior to the fit of γ and t is the mean % of annotated foreground voxels (therefore $t \in [0, 1]$) and $\hat{t_0}$ is it on the starting budget while y is the performance (Mean Dice). y_{full} is the performance on the entire dataset using a trainer with identical length trained on the entire dataset and $\hat{y}(\hat{t_0})$ is the mean performance on the starting budgets.

$$y(t) = (\hat{y}(\hat{t_0}) - \hat{y}_{\text{full}}) \exp(-\gamma(t - \hat{t_0})) + \hat{y}_{\text{full}} \tag{C.1}$$

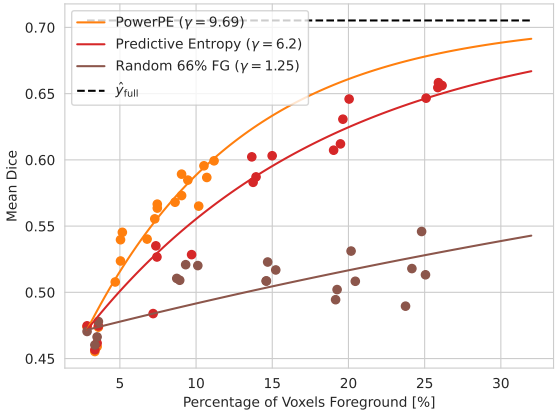


Figure C.1: Visualization of a fit for the FG-Eff on the KiTS Medium-Label Regime showing the QMs: Predictive Entropy, PowerPE and Random 66% FG. The points show the actual performance of all 4 seeds. The γ (FG-Eff) values allow to capture that PowerPE requires much less foreground to achieve a similar performance than Predictive Entropy and also merges the information that even though Random 66% FG and that while Predictive Entropy queries a similar amount of foreground as Random 66% FG, the latter is much less performant.
Fit values: $\hat{t}_0 = 0.028$, $\hat{y}_{\text{full}} = 0.705$, $\hat{y}(\hat{t}_0) = 0.472$

Mathematical Assumptions

- The behavior can be modelled with an exponential decay.
- $y(t) < \hat{y}_{\text{full}} \forall t \in [t_0, t_{\max}]$. Caveat $y(1) = \hat{y}_{\text{full}}$

Interpretation Higher values indicate that a QM is more annotation efficient as it converges faster to the performance obtained when training on the entire dataset. As the number of foreground voxels is a proxy for annotation effort, we also emphasize the importance of evaluating the performance based on the AUBC, Final Dice, and PPM. In a best-case scenario, a QM has a high FG-Eff and excels in the other metrics or is among the better-performing methods.

Generally speaking, a QM which has a high FG-Eff but a very low performance based on all other metrics is not recommended as a good method, as the metric potentially can also be *hacked* by simply querying a very small amount of foreground and a very steep increase in performance relative to the amount of queried foreground.

Limitation The annotation efficiency as a metric is only meaningful when compared on precisely the same model and training with the same starting budget and annotation budget because the estimated values $\hat{y}(\hat{t}_0)$ and $\hat{y}_{\text{full}}$ change resulting γ values substantially. As the number of foreground voxels represents a proxy for annotation effort, the FG-Eff does not replace other performance metrics but should be seen as an extension of them.

Table C.2: Dataset descriptions and configurations for the main study.

Dataset	ACDC	AMOS	KiTS	Hippocampus
# Classes w.o. Background	3	15	3	2
Median Shape	16.5x237x206	237.5x582x582	526x512x512	36x50x35
Used Spacing	2x0.6875x0.6875	5x1.5625x1.5625	0.78125x0.78125x0.78125	1x1x1
# Pool & Training	150	150	225	195
# Validation	50	50	75	65
Budget: Low [# Patches](% Voxels)	150 (0.75%)	200 (0.26%)	200 (0.16%)	100 (6,51%)
Budget: Medium [# Patches](% Voxels)	300(1.50%)	1000 (1.30%)	1000 (0.80%)	200 (13,02%)
Budget: High [# Patches](% Voxels)	450(2.25%)	2500 (3.25%)	2500 (2.00%)	300 (19,54%)
Query Patch Size	4x40x40	32x74x74	60x64x64	20x20x20
$\frac{\text{Query Patch Size}}{\text{\# Voxels Dataset}}$	0.0050%	0.0013%	0.0008%	0.06513%
Test set Mean Dice (1000 Epochs)	0.912	0.893	0.773	0.895
Test set Mean Dice (500 Epochs)	0.912	0.883	0.751	0.895
Test set Mean Dice (200 Epochs)	0.910	0.860	0.705	0.895

C.5 Dataset Details

Key dataset characteristics are shown in table C.2.

ACDC Class names in order of labels (ascending): right ventricle, myocardium, left ventricular cavity

AMOS Class names in order of labels (ascending): spleen, right kidney, left kidney, gall bladder, esophagus, liver, stomach, aorta, postcava, pancreas, right adrenal gland, left adrenal gland, duodenum, bladder, prostate/uterus

Hippocampus Class names in order of labels (ascending): anterior hippocampus, posterior hippocampus

KiTS Class names in order of labels (ascending): kidney, kidney-tumor, kidney-cyst

C.6 Main Study Results

The overall design of the main study, alongside details of the ablation studies, is shown in fig. C.2. The detailed results with regard to AUBC, Final Dice and FG-Eff for each dataset and Label Regime are shown in table C.3.

Further, we show the aggregated PPMs for each dataset separately in fig. C.3.

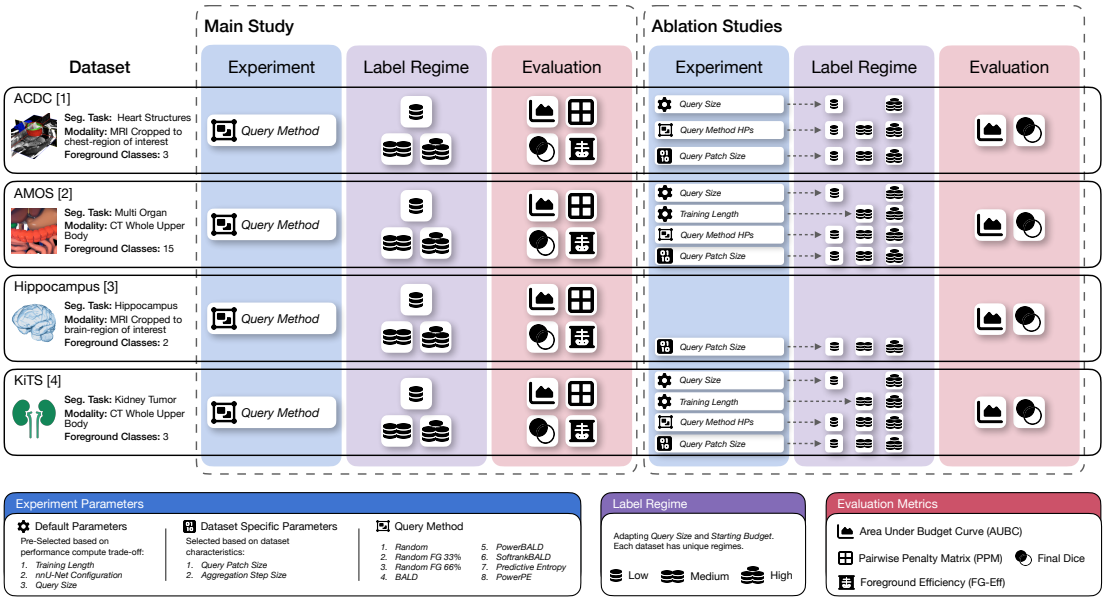


Figure C.2: Systematic schema of our empirical study. It is comprised of one Main Study, which focuses on the evaluation of QMs, and four Ablation Studies analyzing the influence of specific design parameters on AL methods. *Query Method HP's* refers to the Noise strength in Noisy QMs ablation.

C.6.1 AMOS

We show visualization of the queried patches for Predictive Entropy, PowerPE, Random 66% FG and Random on the AMOS Low-Label Regime in fig. C.4.

An investigation of the performance of AL methods by the examples of Predictive Entropy and PowerPE when compared to Random and Random 66% FG are shown in fig. C.5. It clearly shows that the main performance difference stems from a subset of classes which get less well predicted when not queried frequent enough.

C.6.2 KiTS

We show visualization of the queried patches for Predictive Entropy, PowerPE, Random 66% FG and Random on the AMOS Low-Label Regime in fig. C.6.

Table C.3: Fine-Grained Results for the Main Study for each dataset. Higher values are better and colorization goes from bright (best) to dark orange(worst). AUBC and Final Dice are reported with a factor (×100) for improved readability. AUBC, Final and Beta can only directly compared for each Label Regime on each dataset.

(a) ACDC

Dataset	ACDC								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	79.84 ± 0.59	86.44 ± 0.96	26.99 ± 3.11	85.85 ± 0.45	89.62 ± 0.15	21.85 ± 4.16	87.74 ± 0.38	90.47 ± 0.18	14.99 ± 1.14
PowerBALD	81.18 ± 0.58	86.46 ± 0.55	46.30 ± 13.10	85.63 ± 0.37	89.07 ± 0.21	27.69 ± 3.96	87.50 ± 0.44	89.80 ± 0.17	17.83 ± 1.82
SoftrankBALD	80.71 ± 0.92	86.50 ± 0.95	35.72 ± 7.09	85.89 ± 0.49	89.33 ± 0.27	26.28 ± 4.97	87.28 ± 0.68	90.17 ± 0.14	14.44 ± 1.32
Predictive Entropy	80.02 ± 1.54	86.54 ± 0.95	26.50 ± 4.40	85.53 ± 0.59	89.42 ± 0.07	21.11 ± 3.07	87.65 ± 0.27	90.52 ± 0.06	13.50 ± 1.22
PowerPE	80.46 ± 0.30	86.56 ± 0.40	47.89 ± 14.09	85.24 ± 0.69	89.05 ± 0.22	27.86 ± 4.96	87.21 ± 0.60	89.67 ± 0.15	16.44 ± 1.18
Random	76.65 ± 0.81	80.34 ± 1.64	59.28 ± 33.54	82.24 ± 1.25	83.46 ± 0.87	38.10 ± 8.38	84.69 ± 0.96	86.28 ± 1.08	21.45 ± 3.85
Random 33% FG	81.28 ± 0.56	85.09 ± 1.14	40.89 ± 9.71	84.61 ± 0.65	87.51 ± 0.56	21.22 ± 1.48	86.95 ± 0.74	89.06 ± 0.44	15.72 ± 1.42
Random 66% FG	82.32 ± 0.33	86.70 ± 0.48	31.21 ± 4.32	86.16 ± 0.44	88.62 ± 0.52	18.92 ± 2.12	87.86 ± 0.33	89.94 ± 0.09	13.38 ± 0.80

(b) AMOS

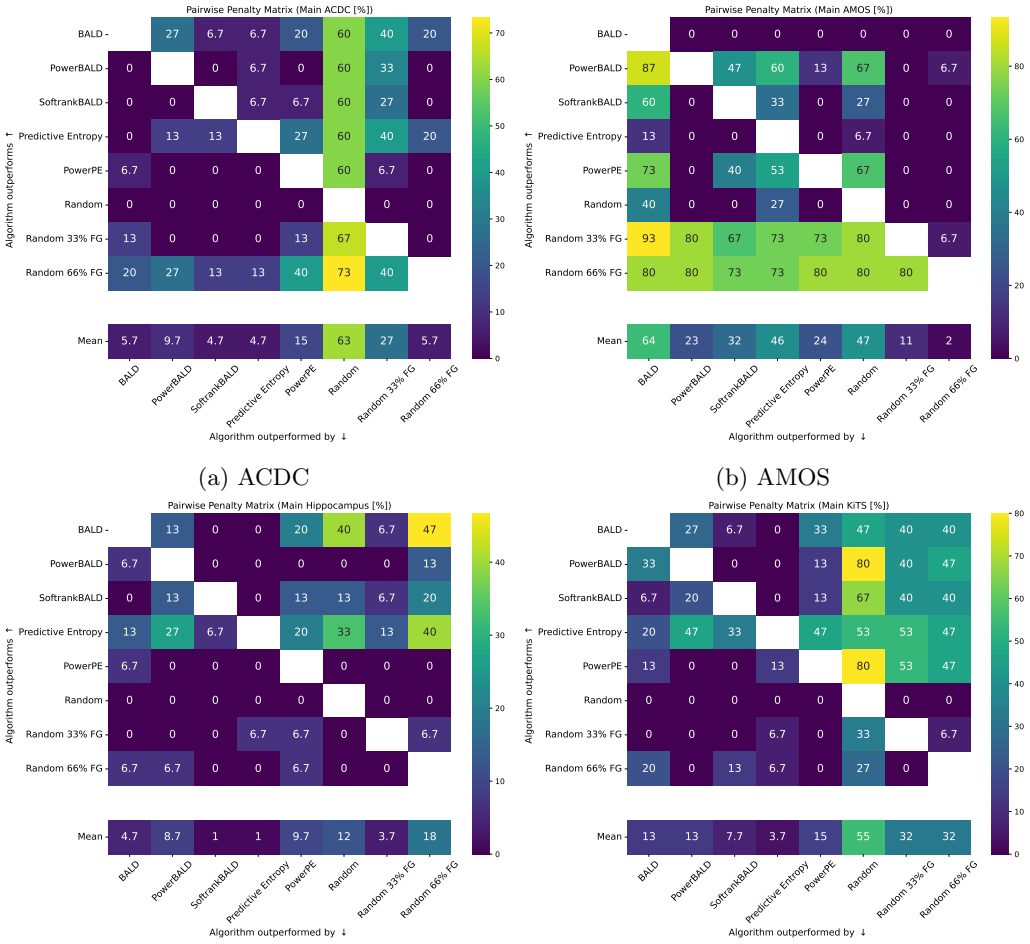
Dataset	AMOS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	38.69 ± 2.34	34.05 ± 1.58	-22.66 ± 8.50	52.56 ± 2.74	59.26 ± 2.73	1.54 ± 0.22	69.38 ± 0.70	74.95 ± 2.38	-0.45 ± 0.20
PowerBALD	50.34 ± 3.00	56.18 ± 1.24	3.65 ± 14.56	66.11 ± 1.47	73.02 ± 2.01	18.28 ± 0.44	77.86 ± 0.14	80.48 ± 0.48	8.80 ± 0.08
SoftrankBALD	44.49 ± 1.56	45.75 ± 0.95	-11.38 ± 4.19	60.01 ± 0.69	66.72 ± 0.65	5.70 ± 0.10	75.29 ± 1.46	81.23 ± 1.18	3.52 ± 0.38
Predictive Entropy	38.02 ± 3.35	39.19 ± 6.79	-17.92 ± 8.49	56.30 ± 1.78	62.07 ± 1.39	2.65 ± 0.17	71.27 ± 1.52	80.79 ± 2.07	1.02 ± 0.41
PowerPE	47.66 ± 2.50	50.04 ± 2.30	-9.80 ± 12.14	66.74 ± 2.80	73.68 ± 0.92	18.60 ± 1.18	77.92 ± 0.29	80.52 ± 0.16	8.87 ± 0.10
Random	42.26 ± 2.55	36.36 ± 2.92	-134.82 ± 89.07	54.65 ± 2.82	56.22 ± 4.61	10.34 ± 3.29	73.82 ± 0.50	75.48 ± 0.37	7.38 ± 0.62
Random 33% FG	58.05 ± 1.54	62.95 ± 1.03	35.45 ± 11.43	71.78 ± 1.16	78.60 ± 0.37	36.58 ± 2.99	79.53 ± 0.38	82.68 ± 0.19	14.44 ± 0.47
Random 66% FG	62.84 ± 1.88	71.11 ± 1.42	43.63 ± 9.82	74.87 ± 0.64	80.72 ± 0.54	32.62 ± 6.15	80.98 ± 0.19	83.81 ± 0.32	12.33 ± 0.43

(c) Hippocampus

Dataset	Hippocampus								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	88.46 ± 0.03	88.87 ± 0.06	9.58 ± 0.97	88.79 ± 0.02	89.18 ± 0.07	4.52 ± 0.06	89.03 ± 0.05	89.42 ± 0.05	3.49 ± 0.12
PowerBALD	88.20 ± 0.08	88.77 ± 0.11	9.21 ± 0.49	88.76 ± 0.04	89.16 ± 0.06	5.55 ± 0.07	88.98 ± 0.07	89.29 ± 0.10	3.90 ± 0.15
SoftrankBALD	88.44 ± 0.11	88.93 ± 0.18	9.61 ± 0.98	88.72 ± 0.08	89.12 ± 0.02	3.90 ± 0.05	89.03 ± 0.06	89.42 ± 0.07	3.60 ± 0.12
Predictive Entropy	88.50 ± 0.06	88.90 ± 0.10	9.75 ± 1.01	88.81 ± 0.04	89.18 ± 0.07	4.23 ± 0.06	89.07 ± 0.07	89.54 ± 0.03	3.74 ± 0.19
PowerPE	88.16 ± 0.08	88.70 ± 0.11	9.25 ± 0.52	88.63 ± 0.09	89.07 ± 0.21	4.41 ± 0.10	88.97 ± 0.07	89.33 ± 0.18	4.08 ± 0.24
Random	88.07 ± 0.10	88.58 ± 0.08	8.76 ± 0.47	88.65 ± 0.11	89.07 ± 0.04	5.10 ± 0.08	88.96 ± 0.09	89.29 ± 0.20	4.41 ± 0.25
Random 33% FG	88.22 ± 0.16	88.70 ± 0.08	9.60 ± 0.81	88.77 ± 0.13	89.22 ± 0.14	6.20 ± 0.17	88.94 ± 0.06	89.33 ± 0.10	3.85 ± 0.15
Random 66% FG	88.28 ± 0.13	88.76 ± 0.14	9.87 ± 0.73	88.63 ± 0.02	89.02 ± 0.04	4.21 ± 0.03	88.92 ± 0.08	89.26 ± 0.06	3.33 ± 0.11

(d) KiTS

Dataset	KiTS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	40.58 ± 2.75	44.03 ± 3.18	7.96 ± 0.82	55.06 ± 1.20	61.97 ± 1.49	6.52 ± 0.14	62.53 ± 0.84	67.57 ± 1.72	9.35 ± 0.46
PowerBALD	45.10 ± 2.91	47.67 ± 3.63	25.24 ± 6.06	54.53 ± 1.40	59.51 ± 1.15	10.18 ± 0.41	61.24 ± 0.57	65.04 ± 0.81	11.89 ± 0.63
SoftrankBALD	42.87 ± 2.91	47.12 ± 3.34	12.41 ± 2.03	54.83 ± 1.79	61.44 ± 2.02	7.00 ± 0.27	62.49 ± 0.74	67.00 ± 0.97	9.82 ± 0.65
Predictive Entropy	40.62 ± 2.74	45.53 ± 3.57	7.04 ± 0.64	57.42 ± 0.54	65.39 ± 0.51	6.20 ± 0.10	64.00 ± 0.15	68.74 ± 0.65	7.83 ± 0.21
PowerPE	45.30 ± 2.05	49.62 ± 1.13	28.70 ± 3.74	54.76 ± 1.10	58.67 ± 1.53	9.69 ± 0.27	60.66 ± 0.66	63.62 ± 1.19	9.60 ± 0.51
Random	38.75 ± 3.36	39.19 ± 4.13	28.46 ± 19.48	47.82 ± 1.84	48.41 ± 1.99	4.10 ± 2.74	53.80 ± 0.68	55.12 ± 1.27	8.85 ± 1.21
Random 33% FG	43.70 ± 0.87	47.35 ± 2.10	16.18 ± 1.32	51.50 ± 1.97	54.08 ± 2.76	3.28 ± 0.15	55.30 ± 1.26	56.79 ± 1.02	1.87 ± 0.04
Random 66% FG	44.97 ± 2.01	46.83 ± 2.53	11.28 ± 1.30	50.78 ± 0.97	51.67 ± 2.31	1.25 ± 0.02	53.73 ± 1.78	55.90 ± 0.84	0.68 ± 0.01



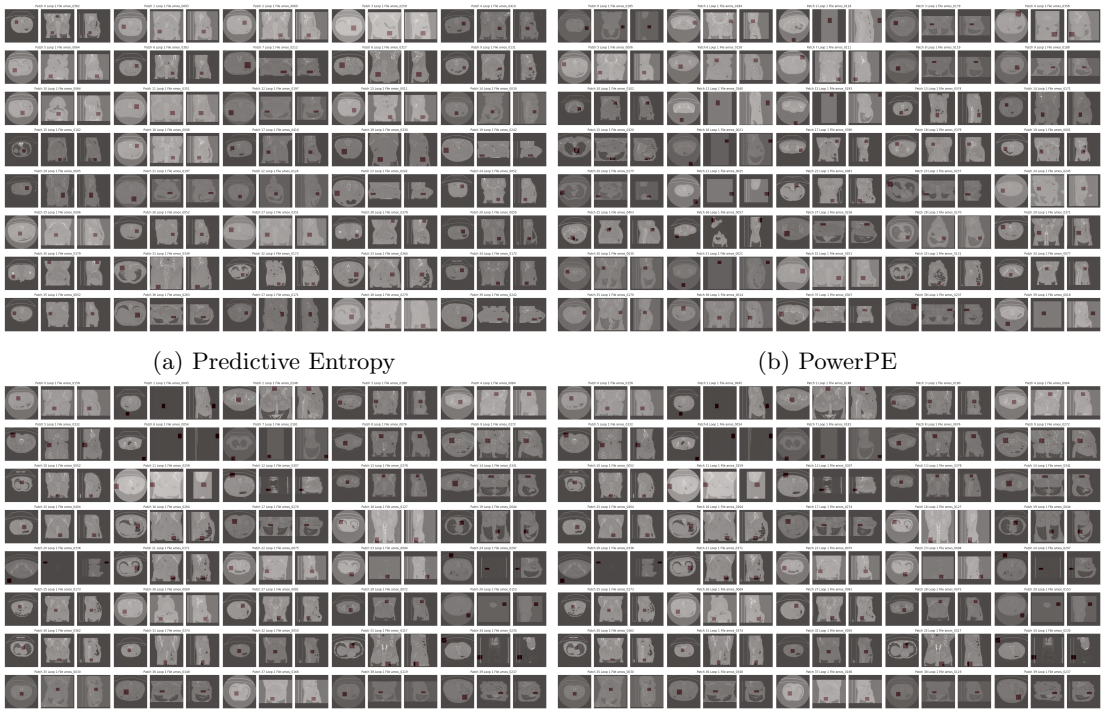
(a) ACDC

(b) AMOS

(c) Hippocampus

(d) KiTS

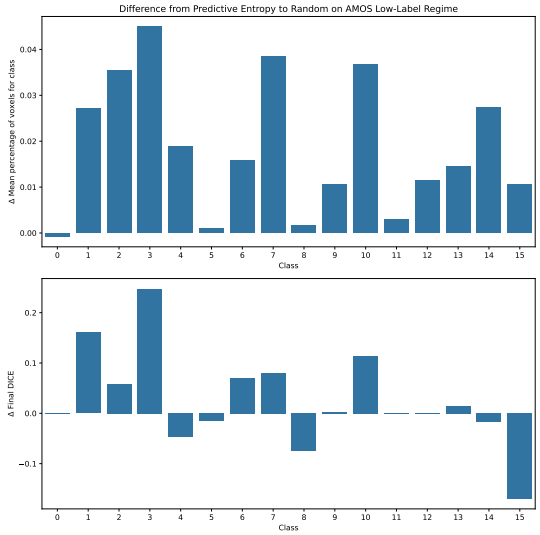
Figure C.3: Pairwise Penalty Matrix aggregated over all Label Regimes for each dataset of the main study.



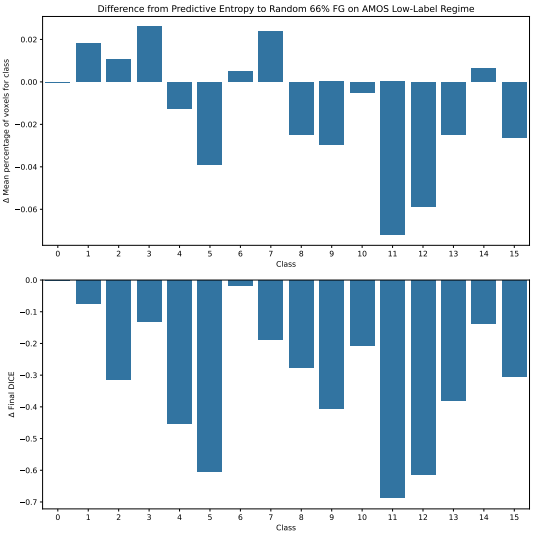
(a) Predictive Entropy (b) PowerPE

(c) Random 66% FG (d) Random

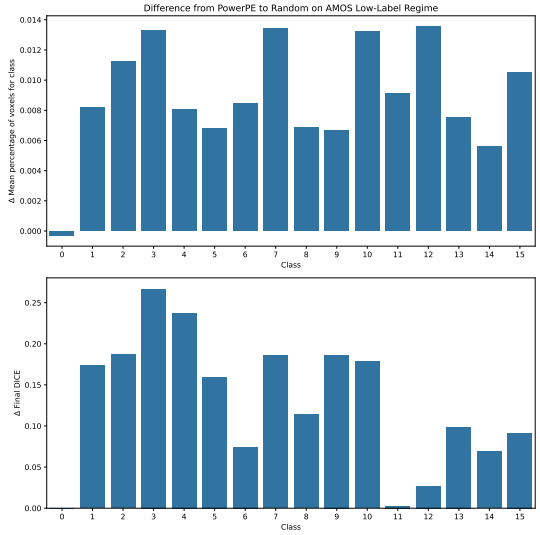
Figure C.4: Queries of the first AL loop on the Low-Label Regime on **AMOS**. Red colored areas are selected patches.
Best viewed on screen with Zoom.
Predictive entropy purely queries regions inside the body with a specific focus on some regions, whereas PowerPE also queries some regions at the borders and is more diverse overall. Random 66% FG queries from multiple regions of the body, but also queries from the outside, and Random queries from quite a substantial amount of regions purely containing air.



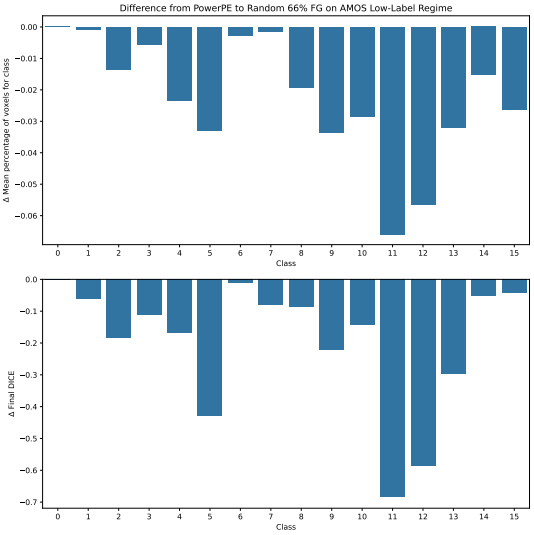
(a) Predictive Entropy



(b) Predictive Entropy

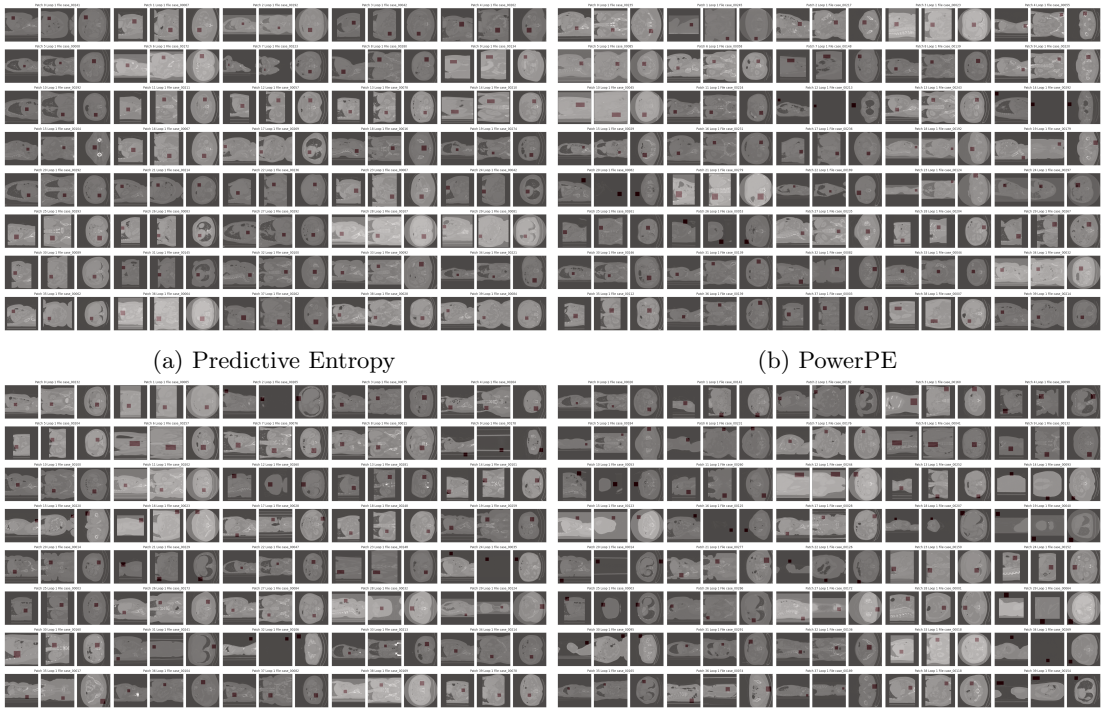


(c) PowerPE



(d) PowerPE

Figure C.5: Visualization of the difference of the percentage of voxels for all classes alongside Final Dice performance on the AMOS Low-Label Regime from Predictive Entropy & PowerPE to Random and Random 66% FG. It shows that less data containing classes 11 & 12 (right & left adrenal gland) is queried by Predictive Entropy and PowerPe (also Random) (5% less of the overall voxels of that class), which is strongly correlated with the Final Dice for these classes being 0. For class 5 (esophagus), a similar behavior can be observed for Predictive Entropy, though not as pronounced. Compared to Predictive Entropy, this effect is weaker for PowerPE, which also queries more data from this class.



(a) Predictive Entropy (b) PowerPE

(c) Random 66% FG (d) Random

Figure C.6: Queries of the first AL loop on the Low-Label Regime on **KiTS**. Red colored areas are selected patches.
Best viewed on screen with Zoom.
Predictive entropy purely queries regions inside the body with a specific focus on the kidneys. In contrast, PowerPE also covers different regions all over the body, still focusing on the kidney, but is more diverse overall. Random 66% FG queries regions in the area of the kidney, but also covers the entire body with some queries containing purely/mostly air. Random queries from quite a substantial number of regions purely containing air.

C.7 Detailed Ablations

Detailed analysis of the ablations can be found in the following subsections:

Ablation 1 Query Size: section C.7.1
Ablation 2 Training Length: section C.7.2
Ablation 3 Noise strength in Noisy QMs: section C.7.3
Ablation 4 Query Patch Size: section C.7.4

An overview of all experiments of the main study and the ablations is given in fig. C.2.

Table C.4: Fine-Grained Results for the query size ablation on ACDC and AMOS. AUBC and Final Dice are reported with a factor (×100) for improved readability. Colors indicate the ranking, darker colors correspond to worse rankings.

(a) ACDC Low Label Regime

Dataset	ACDC					
Setting	QSx1/2		QSx1		QSx2	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	81.16 ± 0.36	87.42 ± 0.49	79.86 ± 1.00	86.44 ± 0.96	78.20 ± 1.87	85.21 ± 1.99
PowerBALD	79.55 ± 0.76	85.35 ± 0.90	80.43 ± 0.25	86.46 ± 0.55	79.08 ± 0.64	85.23 ± 0.31
SoftrankBALD	80.94 ± 1.02	87.13 ± 0.30	80.34 ± 1.05	86.50 ± 0.95	77.98 ± 0.98	85.66 ± 0.81
Predictive Entropy	80.70 ± 0.43	87.30 ± 0.80	79.73 ± 1.40	86.54 ± 0.95	78.32 ± 2.19	86.33 ± 0.46
PowerPE	79.81 ± 0.90	86.08 ± 1.17	79.96 ± 0.44	86.56 ± 0.40	78.81 ± 0.84	85.86 ± 0.28
Random	76.21 ± 1.07	80.61 ± 2.13	75.98 ± 1.29	80.34 ± 1.64	76.97 ± 1.56	81.32 ± 1.61
Random 33% FG	78.89 ± 1.46	83.80 ± 0.76	80.06 ± 0.95	85.09 ± 1.14	79.70 ± 1.01	85.26 ± 0.75
Random 66% FG	80.28 ± 0.56	86.10 ± 0.74	81.16 ± 0.34	86.70 ± 0.48	80.74 ± 0.92	86.52 ± 0.79

(b) ACDC High Label Regime

Dataset	ACDC					
Setting	QSx1/2		QSx1		QSx2	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	87.73 ± 0.55	90.52 ± 0.10	87.54 ± 0.28	90.47 ± 0.18	86.72 ± 0.58	90.38 ± 0.14
PowerBALD	87.36 ± 0.45	89.96 ± 0.25	87.26 ± 0.31	89.80 ± 0.17	86.69 ± 0.59	89.52 ± 0.69
SoftrankBALD	87.45 ± 0.35	90.18 ± 0.10	87.14 ± 0.61	90.17 ± 0.14	86.46 ± 0.45	89.65 ± 0.43
Predictive Entropy	87.93 ± 0.25	90.60 ± 0.12	87.61 ± 0.35	90.52 ± 0.06	86.82 ± 0.52	90.29 ± 0.19
PowerPE	87.27 ± 0.47	89.96 ± 0.19	86.99 ± 0.54	89.67 ± 0.15	86.55 ± 0.83	89.71 ± 0.21
Random	84.85 ± 1.10	86.65 ± 1.30	84.55 ± 0.83	86.28 ± 1.08	84.32 ± 0.98	85.69 ± 1.17
Random 33% FG	86.53 ± 0.68	89.02 ± 0.30	86.75 ± 0.69	89.06 ± 0.44	86.71 ± 0.61	89.28 ± 0.57
Random 66% FG	87.27 ± 0.60	89.79 ± 0.29	87.34 ± 0.25	89.94 ± 0.09	87.34 ± 0.34	90.13 ± 0.20

(c) AMOS Low Label Regime

Dataset	AMOS					
Setting	QSx1/2		QSx1		QSx2	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	38.70 ± 0.71	35.74 ± 2.82	38.86 ± 2.57	34.05 ± 1.58	41.79 ± 2.83	35.92 ± 5.10
PowerBALD	51.99 ± 2.05	55.70 ± 1.24	50.55 ± 3.18	56.18 ± 1.24	48.31 ± 3.13	52.32 ± 2.71
SoftrankBALD	44.30 ± 2.27	45.10 ± 5.16	44.55 ± 0.79	45.75 ± 0.95	41.53 ± 2.29	39.13 ± 5.29
Predictive Entropy	40.98 ± 2.77	41.45 ± 3.97	38.47 ± 3.86	39.19 ± 6.79	38.93 ± 4.05	29.96 ± 4.29
PowerPE	51.58 ± 3.18	54.49 ± 2.81	47.94 ± 3.15	50.04 ± 2.30	46.38 ± 5.15	49.41 ± 6.28
Random	43.47 ± 3.25	40.39 ± 2.87	42.24 ± 2.48	36.36 ± 2.92	40.85 ± 2.70	35.83 ± 2.23
Random 33% FG	54.84 ± 2.83	58.72 ± 2.01	57.64 ± 1.86	62.95 ± 1.03	56.45 ± 1.39	63.51 ± 3.30
Random 66% FG	62.24 ± 1.51	70.96 ± 0.83	62.04 ± 1.57	71.11 ± 1.42	61.96 ± 2.89	71.90 ± 1.58

(d) AMOS High Label Regime

Dataset	AMOS					
Setting	QSx1/2		QSx1		QSx2	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	72.87 ± 1.61	78.70 ± 3.51	69.94 ± 0.55	74.95 ± 2.38	70.67 ± 0.49	69.88 ± 1.31
PowerBALD	77.31 ± 0.22	80.42 ± 0.37	77.51 ± 0.25	80.48 ± 0.48	77.45 ± 0.45	80.55 ± 0.37
SoftrankBALD	75.12 ± 0.84	79.90 ± 1.13	75.11 ± 1.39	81.23 ± 1.18	75.42 ± 1.30	79.86 ± 1.11
Predictive Entropy	75.22 ± 2.04	83.05 ± 0.26	72.06 ± 1.50	80.79 ± 2.07	73.83 ± 1.56	71.98 ± 2.09
PowerPE	77.43 ± 0.67	80.48 ± 0.16	77.36 ± 0.26	80.52 ± 0.16	77.60 ± 0.27	80.29 ± 0.62
Random	73.22 ± 0.88	74.30 ± 1.46	73.95 ± 0.45	75.48 ± 0.37	73.78 ± 0.20	75.44 ± 0.57
Random 33% FG	78.97 ± 0.42	82.37 ± 0.33	79.00 ± 0.32	82.68 ± 0.19	79.21 ± 0.33	82.57 ± 0.12
Random 66% FG	80.15 ± 0.09	83.86 ± 0.17	80.32 ± 0.27	83.81 ± 0.32	80.12 ± 0.39	83.67 ± 0.31

C.7.1 Query Size Ablation

Table C.5: Fine-Grained Results for the query size ablation on KiTS. AUBC and Final Dice are reported with a factor (×100) for improved readability. Colors indicate the ranking, darker colors correspond to worse rankings.

(a) KiTS Low Label Regime

Dataset	KiTS					
Setting	QSx1/2		QSx1		QSx2	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	41.15 ± 2.91	44.04 ± 2.12	40.43 ± 3.18	44.03 ± 3.18	38.47 ± 2.89	39.62 ± 3.95
PowerBALD	44.18 ± 3.62	47.71 ± 4.15	44.29 ± 2.84	47.67 ± 3.63	43.76 ± 2.63	48.13 ± 2.91
SoftrankBALD	42.52 ± 2.76	44.51 ± 1.87	42.91 ± 2.75	47.12 ± 3.34	40.26 ± 3.93	43.25 ± 4.67
Predictive Entropy	42.69 ± 3.58	47.44 ± 4.65	40.17 ± 2.76	45.53 ± 3.57	39.20 ± 2.77	41.81 ± 5.50
PowerPE	45.19 ± 2.73	49.57 ± 3.10	44.64 ± 1.68	49.62 ± 1.13	42.83 ± 2.99	46.39 ± 2.75
Random	38.09 ± 2.22	38.45 ± 3.48	38.71 ± 3.46	39.19 ± 4.13	37.76 ± 3.10	37.98 ± 2.98
Random 33% FG	43.89 ± 0.63	46.99 ± 2.20	43.60 ± 0.92	47.35 ± 2.10	43.97 ± 1.42	48.17 ± 2.32
Random 66% FG	44.48 ± 1.89	48.01 ± 0.90	44.32 ± 2.08	46.83 ± 2.53	43.63 ± 1.56	47.36 ± 0.90

(b) KiTS High Label Regime

Dataset	KiTS					
Setting	QSx1/2		QSx1		QSx2	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	62.85 ± 0.60	68.61 ± 0.58	61.95 ± 1.22	67.57 ± 1.72	61.54 ± 0.59	68.10 ± 0.36
PowerBALD	60.94 ± 0.57	66.00 ± 0.58	60.74 ± 0.49	65.04 ± 0.81	59.54 ± 0.75	64.14 ± 1.23
SoftrankBALD	61.29 ± 1.04	66.78 ± 1.17	61.46 ± 0.90	67.00 ± 0.97	59.91 ± 0.54	66.02 ± 0.51
Predictive Entropy	63.65 ± 0.31	69.04 ± 0.61	63.03 ± 0.70	68.74 ± 0.65	62.10 ± 0.27	68.12 ± 0.80
PowerPE	59.84 ± 0.78	63.48 ± 0.81	59.57 ± 0.69	63.62 ± 1.19	60.16 ± 0.84	64.44 ± 1.04
Random	53.47 ± 1.01	54.45 ± 1.11	53.65 ± 0.64	55.12 ± 1.27	54.09 ± 0.89	55.44 ± 1.79
Random 33% FG	54.30 ± 1.35	56.19 ± 2.66	54.80 ± 0.83	56.79 ± 1.02	54.28 ± 1.72	56.66 ± 1.33
Random 66% FG	53.56 ± 1.11	56.20 ± 1.18	53.92 ± 1.23	55.90 ± 0.84	53.96 ± 0.74	56.60 ± 1.79

C.7.2 Training Length Ablation

We show detailed results for the training length ablation focusing on the ranking in table C.6.

AMOS Training length We observe an especially strong performance increase for longer trained models on AMOS across all AL methods and Random compared to Random 66%FG, which is discussed in section C.6.1. We observe that the longer training leads to substantial performance improvements on classes 11 & 12.

Table C.6: Fine-Grained Results for the training length ablation. AUBC and Final Dice are reported with a factor (×100) for improved readability. Colors indicate the ranking, darker colors correspond to worse rankings.

(a) AMOS Medium Label Regime

Dataset	AMOS					
Setting	200 Epochs		Precomputed		500 Epochs	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	52.56 ± 2.74	59.26 ± 2.73	74.79 ± 1.97	80.01 ± 2.07	75.76 ± 1.20	81.67 ± 2.40
PowerBALD	66.11 ± 1.47	73.02 ± 2.01	78.93 ± 0.58	82.51 ± 0.42	79.87 ± 0.33	83.96 ± 0.41
SoftrankBALD	60.01 ± 0.69	66.72 ± 0.65	78.04 ± 0.34	82.12 ± 0.97	79.04 ± 0.29	83.55 ± 0.08
Predictive Entropy	56.30 ± 1.78	62.07 ± 1.39	77.06 ± 0.61	81.64 ± 0.50	77.21 ± 0.53	83.40 ± 0.41
PowerPE	66.74 ± 2.80	73.68 ± 0.92	79.11 ± 0.30	83.15 ± 0.45	79.27 ± 0.36	83.35 ± 0.21
Random	54.65 ± 2.82	56.22 ± 4.61	72.81 ± 1.31	75.46 ± 0.94	72.81 ± 1.31	75.46 ± 0.94
Random 33% FG	71.78 ± 1.16	78.60 ± 0.37	79.63 ± 0.53	83.70 ± 0.37	79.63 ± 0.53	83.70 ± 0.37
Random 66% FG	74.87 ± 0.64	80.72 ± 0.54	81.31 ± 0.39	84.94 ± 0.46	81.31 ± 0.39	84.94 ± 0.46

(b) AMOS High Label Regime

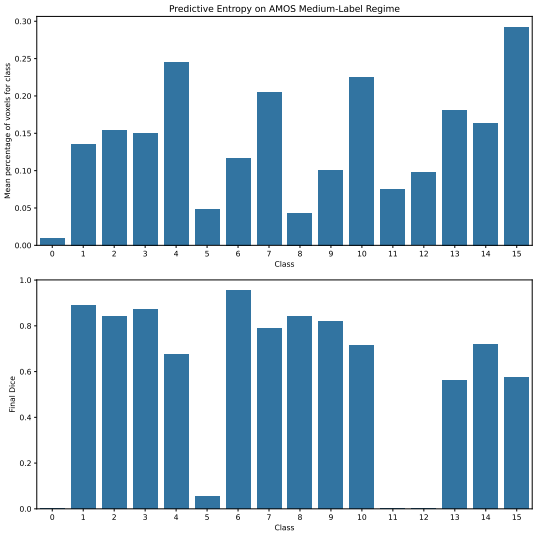
Dataset	AMOS					
Setting	200 Epochs		Precomputed		500 Epochs	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	69.38 ± 0.70	74.95 ± 2.38	83.50 ± 0.17	86.06 ± 0.09	84.37 ± 0.10	87.18 ± 0.07
PowerBALD	77.86 ± 0.14	80.48 ± 0.48	83.98 ± 0.29	85.78 ± 0.06	84.50 ± 0.16	86.35 ± 0.04
SoftrankBALD	75.29 ± 1.46	81.23 ± 1.18	84.28 ± 0.11	86.39 ± 0.10	84.60 ± 0.26	86.83 ± 0.16
Predictive Entropy	71.27 ± 1.52	80.79 ± 2.07	84.00 ± 0.14	86.77 ± 0.13	84.70 ± 0.03	87.52 ± 0.08
PowerPE	77.92 ± 0.29	80.52 ± 0.16	83.86 ± 0.16	85.69 ± 0.19	84.32 ± 0.32	86.23 ± 0.19
Random	73.82 ± 0.50	75.48 ± 0.37	81.31 ± 0.41	82.73 ± 0.06	81.31 ± 0.41	82.73 ± 0.06
Random 33% FG	79.53 ± 0.38	82.68 ± 0.19	84.30 ± 0.13	86.28 ± 0.14	84.30 ± 0.13	86.28 ± 0.14
Random 66% FG	80.98 ± 0.19	83.81 ± 0.32	85.06 ± 0.10	86.98 ± 0.13	85.06 ± 0.10	86.98 ± 0.13

(c) KiTS Medium Label Regime

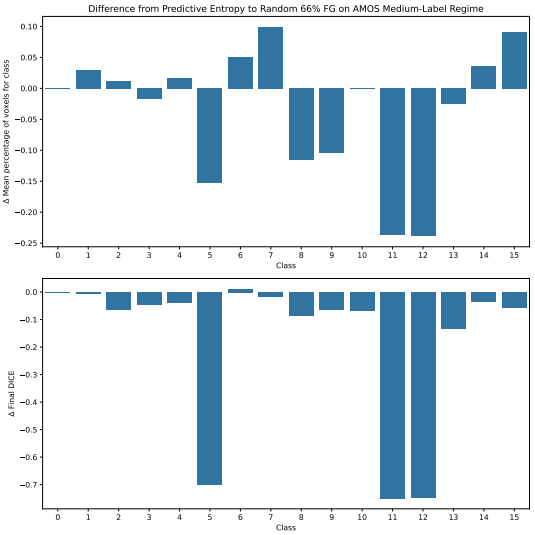
Dataset	KiTS					
Setting	200 Epochs		Precomputed		500 Epochs	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	55.06 ± 1.20	61.97 ± 1.49	61.96 ± 0.66	67.84 ± 0.71	63.69 ± 0.96	70.60 ± 0.55
PowerBALD	54.53 ± 1.40	59.51 ± 1.15	61.76 ± 1.07	65.86 ± 1.17	64.06 ± 0.95	69.78 ± 1.38
SoftrankBALD	54.83 ± 1.79	61.44 ± 2.02	62.44 ± 1.22	68.32 ± 0.38	63.85 ± 1.01	69.80 ± 0.50
Predictive Entropy	57.42 ± 0.54	65.39 ± 0.51	63.79 ± 0.65	69.77 ± 1.10	64.37 ± 0.33	71.83 ± 0.37
PowerPE	54.76 ± 1.10	58.67 ± 1.53	62.22 ± 0.82	66.53 ± 0.39	64.11 ± 0.80	68.71 ± 0.89
Random	47.82 ± 1.84	48.41 ± 1.99	55.35 ± 1.22	56.68 ± 0.91	55.35 ± 1.22	56.68 ± 0.91
Random 33% FG	51.50 ± 1.97	54.08 ± 2.76	59.33 ± 2.33	62.56 ± 3.22	59.33 ± 2.33	62.56 ± 3.22
Random 66% FG	50.78 ± 0.97	51.67 ± 2.31	58.43 ± 1.08	61.27 ± 1.72	58.43 ± 1.08	61.27 ± 1.72

(d) KiTS High Label Regime

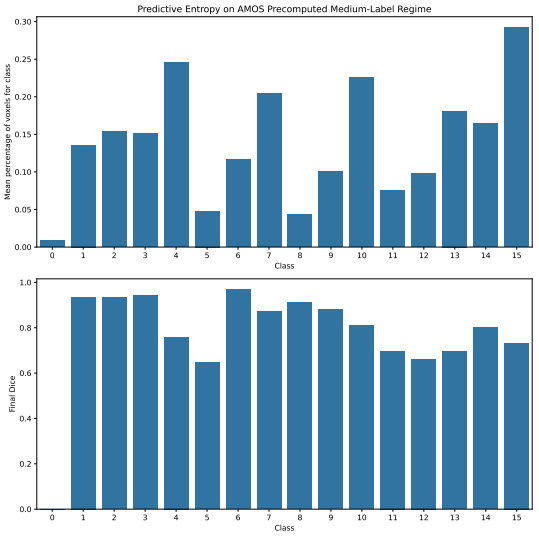
Dataset	KiTS					
Setting	200 Epochs		Precomputed		500 Epochs	
Metric	AUBC	Final Dice	AUBC	Final Dice	AUBC	Final Dice
Query Method						
BALD	62.53 ± 0.84	67.57 ± 1.72	68.83 ± 0.94	72.47 ± 0.46	70.47 ± 0.47	74.51 ± 0.50
PowerBALD	61.24 ± 0.57	65.04 ± 0.81	68.44 ± 0.47	71.47 ± 0.54	69.82 ± 0.50	73.15 ± 0.49
SoftrankBALD	62.49 ± 0.74	67.00 ± 0.97	69.13 ± 0.23	73.44 ± 0.49	70.17 ± 0.62	74.28 ± 0.57
Predictive Entropy	64.00 ± 0.15	68.74 ± 0.65	69.77 ± 0.46	73.78 ± 0.87	70.65 ± 0.31	74.21 ± 0.14
PowerPE	60.66 ± 0.66	63.62 ± 1.19	68.19 ± 0.33	70.48 ± 0.67	69.91 ± 0.25	72.78 ± 0.84
Random	53.80 ± 0.68	55.12 ± 1.27	63.30 ± 1.11	65.36 ± 0.94	63.30 ± 1.11	65.36 ± 0.94
Random 33% FG	55.30 ± 1.26	56.79 ± 1.02	65.27 ± 1.59	68.81 ± 1.08	65.27 ± 1.59	68.81 ± 1.08
Random 66% FG	53.73 ± 1.78	55.90 ± 0.84	64.63 ± 2.52	68.03 ± 0.46	64.63 ± 2.52	68.03 ± 0.46



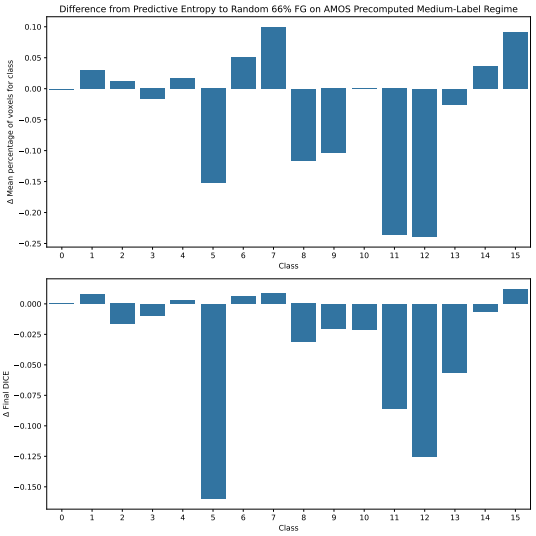
(a) 200 Epochs (absolute)



(b) 200 Epochs (difference)



(c) Precomputed (absolute)



(d) Precomputed (difference)

Figure C.7: Visualization of absolute values and the difference of the percentage of voxels for all classes alongside Final Dice performance on the AMOS Medium-Label Regime from Predictive Entropy. It shows that less data containing classes 11 & 12 (right & left adrenal gland) is queried by Predictive Entropy (also Random) (5% less of the overall voxels of that class), which is strongly correlated with the Final Dice for these classes being 0 for the 200 Epochs results (a) whereas it is at ≈ 0.7 for the Precomputed models on the exact same data just trained for 500 epochs (c). This substantially reduces the performance gap compared to Random 66% FG as can be seen in (b &d).

Table C.7: Ablating the influence of the noise parameter for PowerBALD. AUBC and Final Dice are reported with a factor (×100) for improved readability. The values leading to the highest AUBC and Final Mean Dice increase for larger budgets across all datasets.

(a) ACDC

Dataset	ACDC								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
PowerBALD (b=1)	81.18 ± 0.58	86.46 ± 0.55	46.30 ± 13.10	85.63 ± 0.37	89.07 ± 0.21	27.69 ± 3.96	87.50 ± 0.44	89.80 ± 0.17	17.83 ± 1.82
PowerBALD (b=5)	80.89 ± 1.11	87.29 ± 0.34	36.03 ± 6.27	86.21 ± 0.39	89.53 ± 0.35	28.16 ± 6.55	87.45 ± 0.54	90.40 ± 0.34	13.31 ± 1.83
PowerBALD (b=10)	81.26 ± 1.25	86.74 ± 0.38	33.98 ± 5.87	85.65 ± 0.65	89.58 ± 0.23	22.05 ± 3.47	87.84 ± 0.46	90.46 ± 0.20	15.80 ± 1.85
PowerBALD (b=20)	81.45 ± 1.11	86.91 ± 0.79	33.60 ± 8.86	85.63 ± 0.31	89.60 ± 0.14	21.19 ± 3.06	87.62 ± 0.46	90.28 ± 0.20	14.75 ± 1.35
PowerBALD (b=40)	80.50 ± 1.36	86.31 ± 0.34	27.86 ± 4.46	85.39 ± 0.75	89.43 ± 0.26	19.28 ± 3.46	87.60 ± 0.31	90.51 ± 0.19	13.05 ± 0.64
PowerBALD (b=∞)	79.84 ± 0.59	86.44 ± 0.96	26.99 ± 3.11	85.85 ± 0.45	89.62 ± 0.15	21.85 ± 4.16	87.74 ± 0.38	90.47 ± 0.18	14.99 ± 1.14

(b) AMOS

Dataset	AMOS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
PowerBALD (b=1)	50.34 ± 3.00	56.18 ± 1.24	3.65 ± 14.56	66.11 ± 1.47	73.02 ± 2.01	18.28 ± 0.44	77.86 ± 0.14	80.48 ± 0.48	8.80 ± 0.08
PowerBALD (b=5)	47.72 ± 1.70	49.61 ± 1.31	-4.72 ± 3.70	60.87 ± 0.97	66.76 ± 1.09	6.78 ± 0.13	76.11 ± 1.65	82.58 ± 0.60	4.56 ± 0.57
PowerBALD (b=10)	42.89 ± 0.29	42.32 ± 3.03	-13.04 ± 4.86	56.58 ± 2.17	63.99 ± 0.66	4.06 ± 0.17	72.39 ± 1.69	80.71 ± 2.03	1.74 ± 0.49
PowerBALD (b=20)	42.08 ± 2.58	39.77 ± 2.02	-16.51 ± 4.56	55.68 ± 1.62	64.04 ± 2.21	3.40 ± 0.18	70.05 ± 1.00	75.65 ± 4.15	-0.22 ± 0.27
PowerBALD (b=40)	39.82 ± 3.50	37.98 ± 6.99	-19.27 ± 12.73	52.37 ± 1.83	58.76 ± 1.76	1.54 ± 0.17	70.31 ± 1.00	76.62 ± 4.52	0.05 ± 0.30
PowerBALD (b=∞)	38.69 ± 2.34	34.05 ± 1.58	-22.66 ± 8.50	52.56 ± 2.74	59.26 ± 2.73	1.54 ± 0.22	69.38 ± 0.70	74.95 ± 2.38	-0.45 ± 0.20

(c) KiTS

Dataset	KiTS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
PowerBALD (b=1)	45.10 ± 2.91	47.67 ± 3.63	25.24 ± 6.06	54.53 ± 1.40	59.51 ± 1.15	10.18 ± 0.41	61.24 ± 0.57	65.04 ± 0.81	11.89 ± 0.63
PowerBALD (b=5)	43.03 ± 3.65	46.20 ± 4.98	14.30 ± 2.54	54.83 ± 1.30	63.02 ± 1.43	7.93 ± 0.24	62.62 ± 1.09	68.17 ± 0.36	10.56 ± 0.36
PowerBALD (b=10)	42.35 ± 3.72	46.00 ± 3.58	11.83 ± 1.95	54.73 ± 1.70	63.63 ± 1.06	7.44 ± 0.23	63.19 ± 0.38	68.34 ± 0.57	9.54 ± 0.15
PowerBALD (b=20)	41.66 ± 4.43	44.56 ± 6.96	10.96 ± 2.12	55.26 ± 1.63	63.12 ± 0.22	7.20 ± 0.17	62.60 ± 0.57	67.95 ± 0.70	9.46 ± 0.28
PowerBALD (b=40)	39.31 ± 4.50	43.40 ± 4.69	6.93 ± 1.93	54.90 ± 1.20	62.01 ± 1.46	6.54 ± 0.14	62.51 ± 0.63	68.25 ± 0.61	9.21 ± 0.39
PowerBALD (b=∞)	40.58 ± 2.75	44.03 ± 3.18	7.96 ± 0.82	55.06 ± 1.20	61.97 ± 1.49	6.52 ± 0.14	62.53 ± 0.84	67.57 ± 1.72	9.35 ± 0.46

C.7.3 Noise strength in Noisy QMs Ablation

Table C.8: Fine-Grained Results for the patch ablation with setting Patch$\times\frac{1}{2}$ for each dataset. Higher values are better and colorization goes from bright (best) to dark orange(worst). Final Dice is reported with a factor ($\times$100) for improved readability. AUBC, Final and Beta can only directly compared for each Label Regime on each dataset.

(a) ACDC

Dataset	ACDC								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	68.23 ± 2.31	77.72 ± 1.46	230.94 ± 445.25	75.80 ± 1.10	82.59 ± 1.24	149.13 ± 432.05	79.59 ± 1.05	83.99 ± 0.85	75.63 ± 69.85
PowerBALD	65.90 ± 5.61	75.24 ± 2.32	306.70 ± 948.85	77.07 ± 1.11	83.01 ± 1.21	207.23 ± 491.64	80.27 ± 1.39	84.54 ± 1.25	121.75 ± 194.86
SoftrankBALD	66.81 ± 3.68	75.98 ± 0.13	245.78 ± 446.20	76.84 ± 1.31	82.16 ± 0.47	198.50 ± 456.74	79.69 ± 1.03	84.03 ± 1.56	118.55 ± 183.10
Predictive Entropy	65.27 ± 2.45	75.79 ± 2.45	185.36 ± 203.07	74.67 ± 1.26	81.18 ± 1.32	119.08 ± 160.70	79.25 ± 0.95	83.58 ± 1.33	85.12 ± 103.40
PowerPE	65.70 ± 3.90	74.46 ± 2.28	301.78 ± 894.94	76.26 ± 2.36	82.16 ± 2.15	211.14 ± 448.40	79.85 ± 1.31	84.48 ± 1.55	133.05 ± 271.55
Random	59.38 ± 5.56	65.19 ± 4.17	480.79 ± 2317.97	70.99 ± 3.17	76.66 ± 1.22	459.67 ± 713.79	76.30 ± 0.80	79.09 ± 0.46	261.56 ± 531.48
Random 33% FG	67.98 ± 1.51	77.43 ± 0.27	216.59 ± 97.30	75.09 ± 1.78	81.65 ± 1.01	127.48 ± 64.58	79.55 ± 1.12	84.44 ± 0.32	88.02 ± 27.63
Random 66% FG	64.33 ± 1.17	73.97 ± 0.55	101.88 ± 47.08	74.69 ± 0.28	82.18 ± 1.52	71.88 ± 20.74	80.33 ± 0.56	85.88 ± 0.64	56.98 ± 8.47

Table C.9: AMOS

Dataset	AMOS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	13.98 ± 1.24	10.96 ± 2.19	-149.85 ± 369.03	16.93 ± 2.33	17.85 ± 4.60	-10.43 ± 20.83	30.15 ± 1.72	27.72 ± 0.83	-19.51 ± 3.37
PowerBALD	14.54 ± 2.70	11.74 ± 2.59	-247.20 ± 763.03	21.71 ± 1.48	25.83 ± 2.25	8.77 ± 16.48	40.14 ± 1.86	42.40 ± 1.47	8.16 ± 4.96
SoftrankBALD	13.63 ± 2.69	11.39 ± 1.68	-127.00 ± 355.48	19.95 ± 1.55	23.48 ± 3.19	-0.92 ± 8.31	35.13 ± 2.40	39.37 ± 1.87	-3.29 ± 5.42
Predictive Entropy	13.83 ± 2.12	12.28 ± 1.98	-83.38 ± 257.65	24.61 ± 2.34	27.37 ± 5.21	7.05 ± 2.20	36.63 ± 6.19	43.86 ± 6.88	1.59 ± 4.86
PowerPE	15.18 ± 2.85	13.00 ± 4.65	-210.89 ± 558.96	23.28 ± 1.26	27.05 ± 2.03	14.76 ± 8.90	43.20 ± 1.78	47.34 ± 3.06	18.40 ± 3.35
Random	12.78 ± 2.02	8.89 ± 1.91	-937.96 ± 5770.12	16.14 ± 1.62	16.99 ± 3.32	-91.07 ± 168.72	37.56 ± 1.57	37.28 ± 3.44	-4.41 ± 22.82
Random 33% FG	22.10 ± 1.18	24.14 ± 4.37	15.47 ± 104.65	39.32 ± 2.68	51.61 ± 4.21	103.40 ± 24.63	56.68 ± 1.86	65.54 ± 1.82	63.35 ± 10.78
Random 66% FG	31.10 ± 2.19	39.70 ± 0.34	135.25 ± 66.74	48.12 ± 0.68	60.25 ± 0.45	94.66 ± 28.61	62.07 ± 0.73	70.71 ± 0.60	51.43 ± 8.39

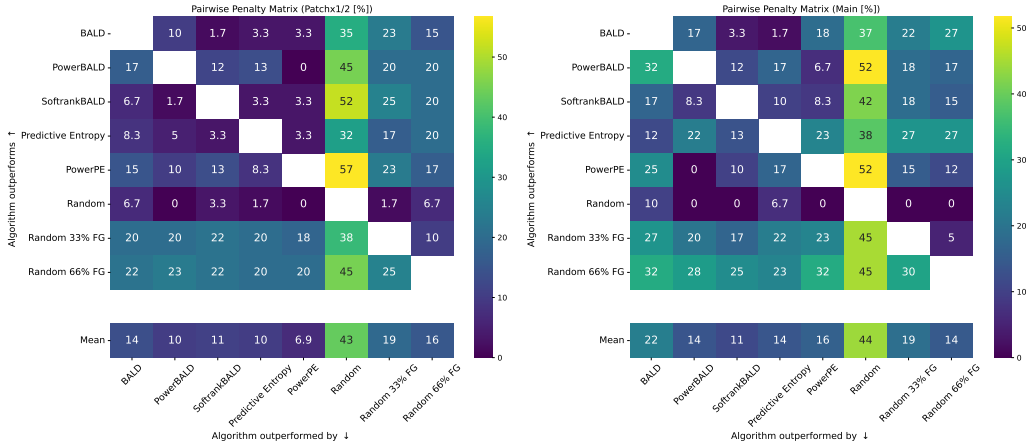
(a) Hippocampus

Dataset	Hippocampus								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	86.42 ± 0.47	87.85 ± 0.15	72.53 ± 176.66	87.64 ± 0.17	88.43 ± 0.19	15.03 ± 2.49	87.99 ± 0.16	88.76 ± 0.08	12.55 ± 1.38
PowerBALD	86.07 ± 0.35	87.45 ± 0.37	79.38 ± 99.29	87.32 ± 0.04	88.12 ± 0.04	18.16 ± 1.27	87.82 ± 0.04	88.47 ± 0.07	17.41 ± 1.33
SoftrankBALD	86.44 ± 0.33	87.66 ± 0.32	73.37 ± 156.28	87.54 ± 0.17	88.27 ± 0.10	16.64 ± 2.19	87.92 ± 0.07	88.66 ± 0.14	15.25 ± 1.73
Predictive Entropy	86.34 ± 0.22	87.69 ± 0.09	63.90 ± 130.04	87.43 ± 0.14	88.41 ± 0.09	13.37 ± 1.22	87.99 ± 0.14	88.74 ± 0.09	12.40 ± 1.77
PowerPE	86.21 ± 0.70	87.56 ± 0.51	84.64 ± 146.26	87.43 ± 0.11	88.29 ± 0.11	19.82 ± 2.64	87.94 ± 0.11	88.43 ± 0.15	18.24 ± 2.48
Random	85.62 ± 0.65	86.74 ± 0.31	118.84 ± 225.57	87.06 ± 0.21	87.76 ± 0.10	25.66 ± 5.49	87.58 ± 0.15	88.13 ± 0.15	26.30 ± 3.24
Random 33% FG	85.69 ± 0.56	87.02 ± 0.19	79.09 ± 126.53	87.26 ± 0.17	88.00 ± 0.07	17.20 ± 2.50	87.74 ± 0.06	88.31 ± 0.11	15.53 ± 1.30
Random 66% FG	86.24 ± 0.13	87.54 ± 0.15	57.33 ± 56.56	87.49 ± 0.21	88.27 ± 0.10	14.92 ± 1.15	87.85 ± 0.21	88.54 ± 0.17	12.51 ± 0.66

(b) KiTS

Dataset	KiTS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	25.10 ± 0.55	31.76 ± 4.51	87.05 ± 97.39	38.56 ± 3.27	43.25 ± 3.79	30.75 ± 9.84	48.46 ± 1.19	53.50 ± 1.28	24.01 ± 6.81
PowerBALD	27.91 ± 1.74	29.39 ± 1.30	185.27 ± 580.70	41.70 ± 1.09	45.59 ± 1.40	77.67 ± 43.37	49.60 ± 0.95	54.00 ± 1.25	43.11 ± 17.12
SoftrankBALD	25.67 ± 2.68	31.47 ± 1.54	90.97 ± 90.60	41.08 ± 1.00	46.14 ± 1.77	45.78 ± 16.15	49.08 ± 1.12	54.39 ± 1.53	31.79 ± 10.31
Predictive Entropy	24.08 ± 1.56	29.07 ± 5.82	41.91 ± 25.47	40.99 ± 3.00	46.80 ± 3.63	23.66 ± 2.82	50.22 ± 1.42	55.79 ± 1.07	14.88 ± 1.68
PowerPE	27.96 ± 3.53	30.88 ± 4.84	208.04 ± 653.54	42.26 ± 0.77	46.55 ± 0.95	82.17 ± 50.00	49.48 ± 1.57	53.59 ± 1.03	44.22 ± 21.94
Random	22.00 ± 1.62	22.85 ± 2.15	140.75 ± 532.82	35.14 ± 2.00	37.95 ± 1.74	90.43 ± 136.75	42.73 ± 1.09	44.35 ± 1.60	47.22 ± 74.17
Random 33% FG	23.88 ± 3.43	28.83 ± 1.46	49.19 ± 31.52	37.88 ± 0.74	41.24 ± 0.88	17.18 ± 1.39	42.28 ± 1.19	44.19 ± 1.31	3.51 ± 0.15
Random 66% FG	24.43 ± 1.96	28.80 ± 2.90	29.92 ± 7.79	34.12 ± 1.40	36.52 ± 1.24	4.17 ± 0.26	40.24 ± 1.31	42.58 ± 0.88	0.69 ± 0.04

C.7.4 Query Patch Size Ablation



(a) Patchx1/2 (b) Patchx1

Figure C.8: PPM aggregated over all Label Regimes for each dataset for the Patch Size Ablation with size Patchx1/2 and Patchx1 (Main Study).

Table C.10: ACDC Mean Ranks

	Rank Mean Dice AUBC		Rank Mean Dice Final	
Setting	Main	Patchx1-2	Main	Patchx1-2
Query Method				
BALD	3.83	3.25	2.50	2.92
PowerBALD	3.67	2.67	4.58	2.92
SoftrankBALD	3.92	3.17	3.25	4.33
Predictive Entropy	4.50	6.08	2.17	5.75
PowerPE	5.58	3.83	4.42	4.50
Random	8.00	8.00	8.00	8.00
Random 33% FG	5.42	4.08	7.00	3.58
Random 66% FG	1.08	4.92	4.08	4.00

Table C.11: AMOS Mean Ranks

	Rank Mean Dice AUBC		Rank Mean Dice Final	
Setting	Main	Patchx1-2	Main	Patchx1-2
Query Method				
BALD	7.75	6.92	7.58	7.17
PowerBALD	3.58	4.50	4.08	4.92
SoftrankBALD	5.00	6.42	4.42	5.83
Predictive Entropy	6.92	4.83	5.42	4.00
PowerPE	3.42	3.33	4.08	3.50
Random	6.33	7.00	7.42	7.58
Random 33% FG	2.00	2.00	2.00	2.00
Random 66% FG	1.00	1.00	1.00	1.00

Table C.12: Hippocampus Mean Ranks

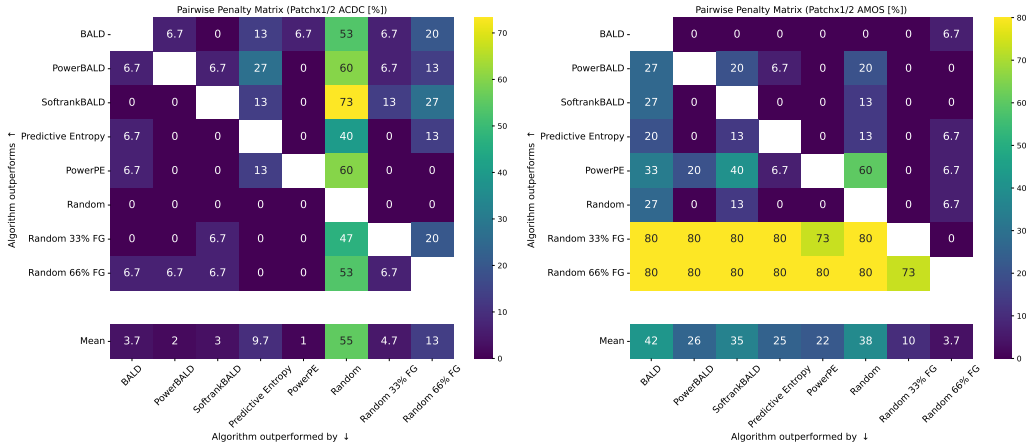
	Rank Mean Dice AUBC		Rank Mean Dice Final	
Setting	Main	Patchx1-2	Main	Patchx1-2
Query Method				
BALD	2.33	1.67	2.67	1.17
PowerBALD	4.75	5.83	4.83	5.50
SoftrankBALD	3.33	2.67	3.00	3.08
Predictive Entropy	1.08	3.00	1.83	2.25
PowerPE	6.17	3.92	5.92	4.58
Random	6.83	7.92	6.83	8.00
Random 33% FG	5.17	7.00	4.33	7.00
Random 66% FG	6.33	4.00	6.58	4.42

Table C.13: KiTS Mean Ranks

	Rank Mean Dice AUBC		Rank Mean Dice Final	
Setting	Main	Patchx1-2	Main	Patchx1-2
Query Method				
BALD	3.92	4.83	3.83	3.83
PowerBALD	3.50	1.92	3.58	3.75
SoftrankBALD	3.58	3.58	3.00	2.33
Predictive Entropy	2.75	3.58	2.58	2.67
PowerPE	3.50	1.67	3.67	3.08
Random	7.83	7.08	8.00	7.17
Random 33% FG	5.42	6.25	5.17	6.00
Random 66% FG	5.50	7.08	6.17	7.17

Table C.14: Average Mean Ranks over all datasets

	Rank Mean Dice AUBC		Rank Mean Dice Final	
Setting	Main	Patchx1-2	Main	Patchx1-2
Query Method				
BALD	4.46	4.17	4.15	3.77
PowerBALD	3.88	3.73	4.27	4.27
SoftrankBALD	3.96	3.96	3.42	3.90
Predictive Entropy	3.81	4.38	3.00	3.67
PowerPE	4.67	3.19	4.52	3.92
Random	7.25	7.50	7.56	7.69
Random 33% FG	4.50	4.83	4.62	4.65
Random 66% FG	3.48	4.25	4.46	4.15



(a) ACDC

(b) AMOS

(c) Hippocampus

(d) KiTS

Figure C.9: Pairwise Penalty Matrix aggregated over all Label Regimes for each dataset for the Patch Size Ablation with size Patchx1/2.

C.8 Leave-One-Out Analysis of Rankings on the Main Study

We additionally analyze the results of the main study shown in section C.6 by means of computing the rankings for AUBC and Final Dice in a leave-one-out fashion based on experimental seeds.

Results Alternative versions of the main overview figure (shown in fig. 7.4) which are obtained by means of aggregating the mean rank for each scenario from the 4 leave-one-out rankings, are shown for the AUBC in fig. C.10 and for the Final Dice in fig. C.11.

Detailed results showing also the distribution of the four obtained rankings are shown for the AUBC in fig. C.12 and for the Final Dice in fig. C.13.

Take-Away: General groups of ranking performance of QMs can be observed in all scenarios where certain groups of QMs are better than others. Overall, based on this analysis, little overall changes compared to the ranking shown in fig. 7.4 are observed.

Details Each experiment is performed with 4 different seeds, therefore each ranking is obtained 4 times.

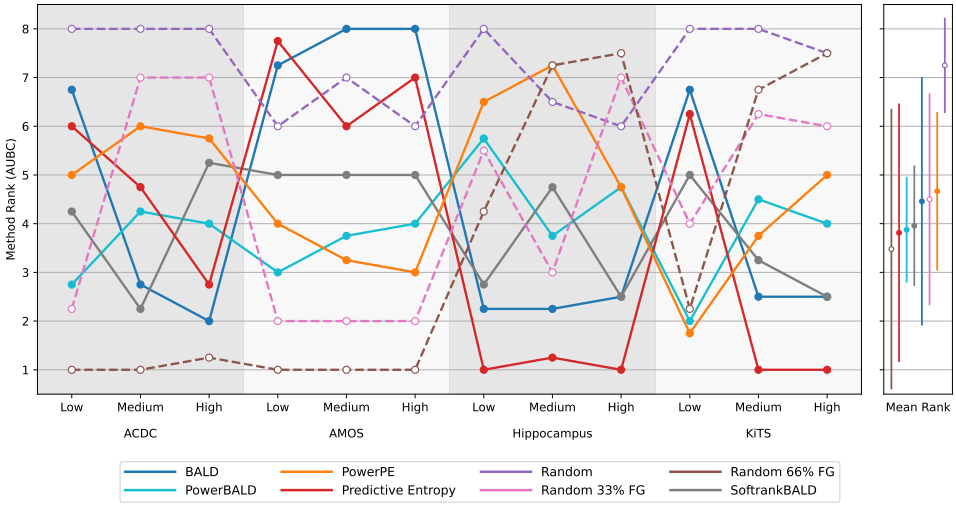


Figure C.10: **Leave-One-Out Overview of Main Study Overview for AUBC.** Ranking of methods according to AUBC for each dataset and its Label Regimes (Low, Medium & High) alongside mean with standard deviations (bar).

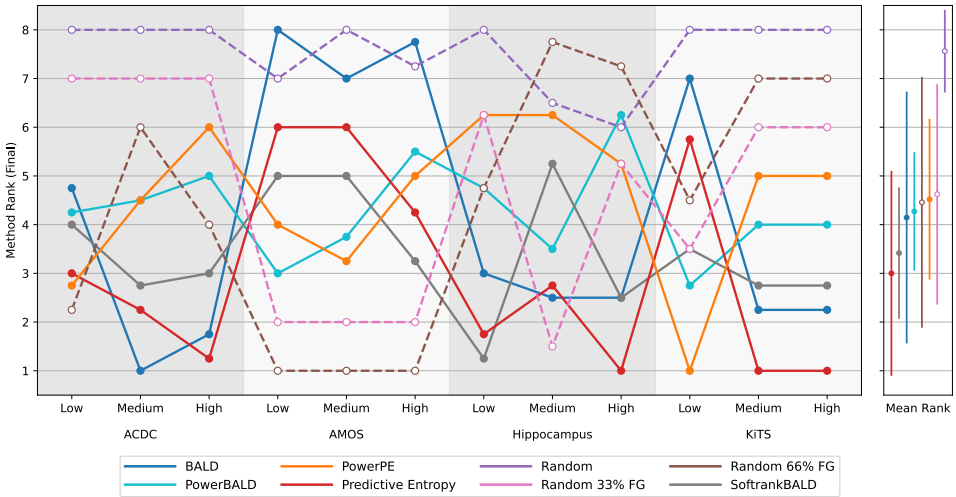


Figure C.11: **Leave-One-Out Overview of Main Study for Final Dice.** Ranking of methods according to Final Dice for each dataset and its Label Regimes (Low, Medium & High) alongside mean with standard deviations (bar).

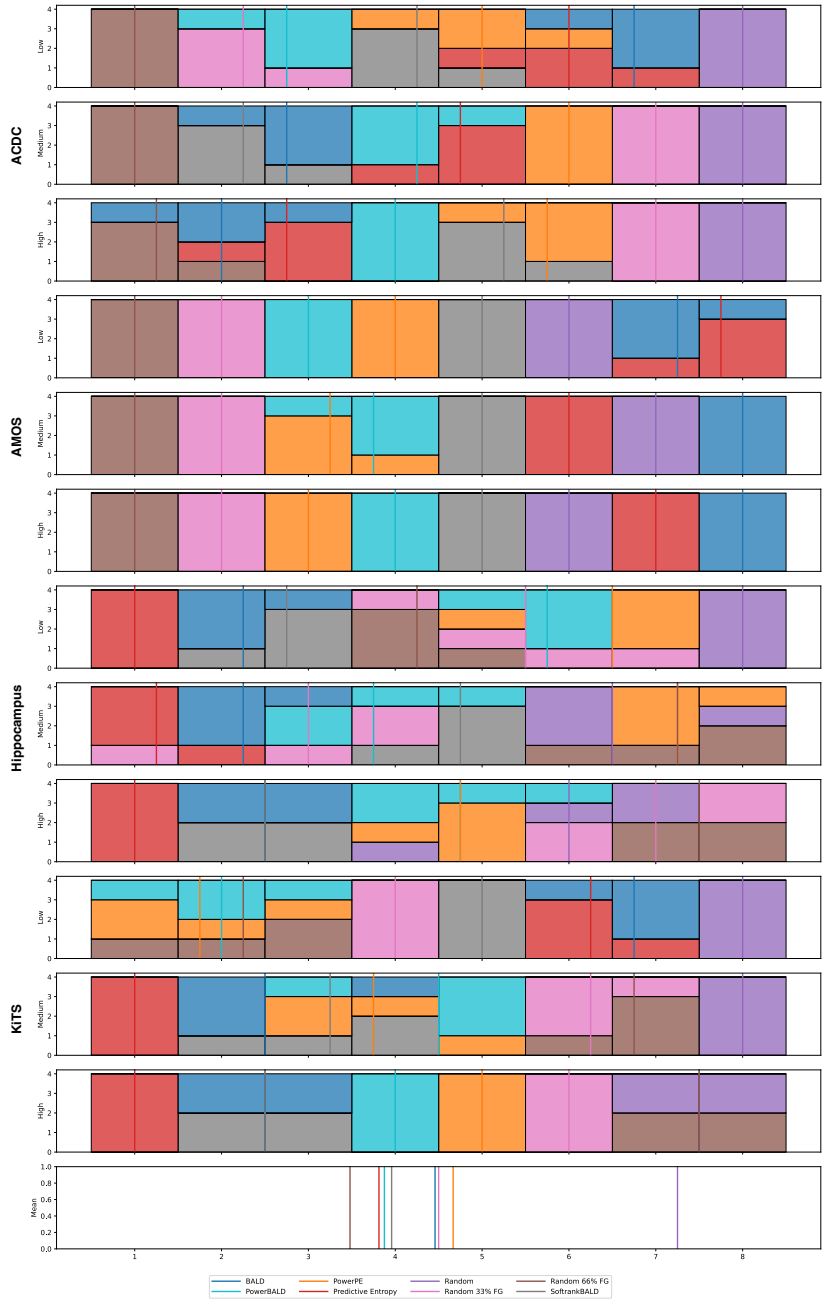


Figure C.12: **Leave-One-Out Detailed Results of Main Study for AUBC.** Ranking of methods according to AUBC for each dataset and its Label Regimes (Low, Medium & High). A specific colored field of height 1 at x-axis x denotes that for one of the four seeds the method corresponding to this color obtained in the leave-one-out (seed based) ranking place x.

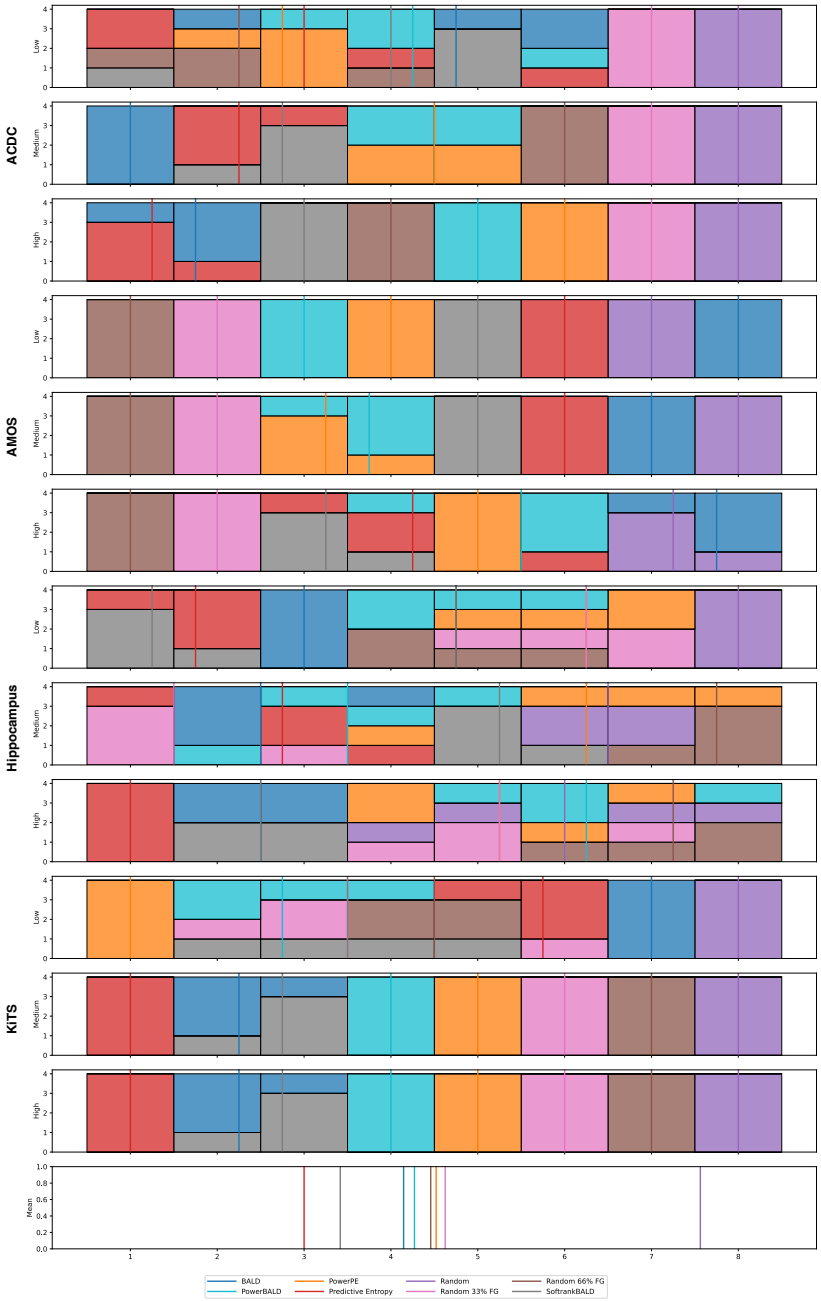


Figure C.13: **Leave-One-Out Detailed Results of Main Study for Final Dice.** Ranking of methods according to AUBC for each dataset and its Label Regimes (Low, Medium & High). A specific colored field of height 1 at x-axis x denotes that for one of the four seeds the method corresponding to this color obtained in the leave-one-out (seed based) ranking place x.

C.9 Model Prediction Visualizations

We provide exemplary visualizations of the predicted segmentation masks for different QMs for ACDC (fig. C.14), AMOS (fig. C.15), Hippocampus (fig. C.16), and KiTS (fig. C.17) of the Main Study. We selected the following model configurations:

- ACDC (fig. C.14): Low-Label setting of the main study (annotation budget: 150, query patch size: $4 \times 40 \times 40$); seed: 12347
- AMOS (fig. C.15): Low-Label setting of the main study (annotation budget: 200, query patch size: $32 \times 74 \times 74$); seed: 12347
- Hippocampus (fig. C.16): Low-Label setting of the main study (annotation budget: 100, query patch size: $20 \times 20 \times 20$); seed: 12345
- KiTS (fig. C.17): Low-Label setting of the main study (annotation budget: 200, query patch size: $64 \times 64 \times 64$); seed: 12347

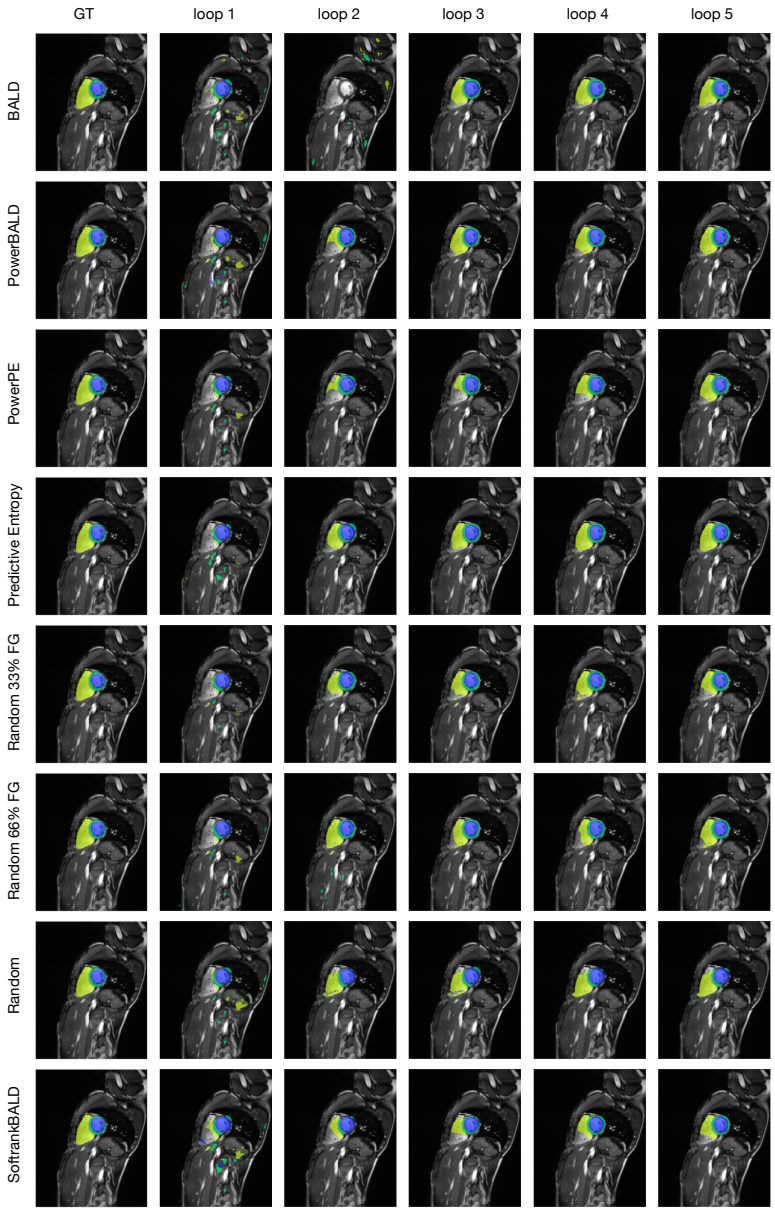


Figure C.14: **Exemplary Model Predictions on ACDC for different QMs.** Column 1: Ground Truth (GT) segmentation masks; Column 2-6: predicted segmentations after each AL loop.

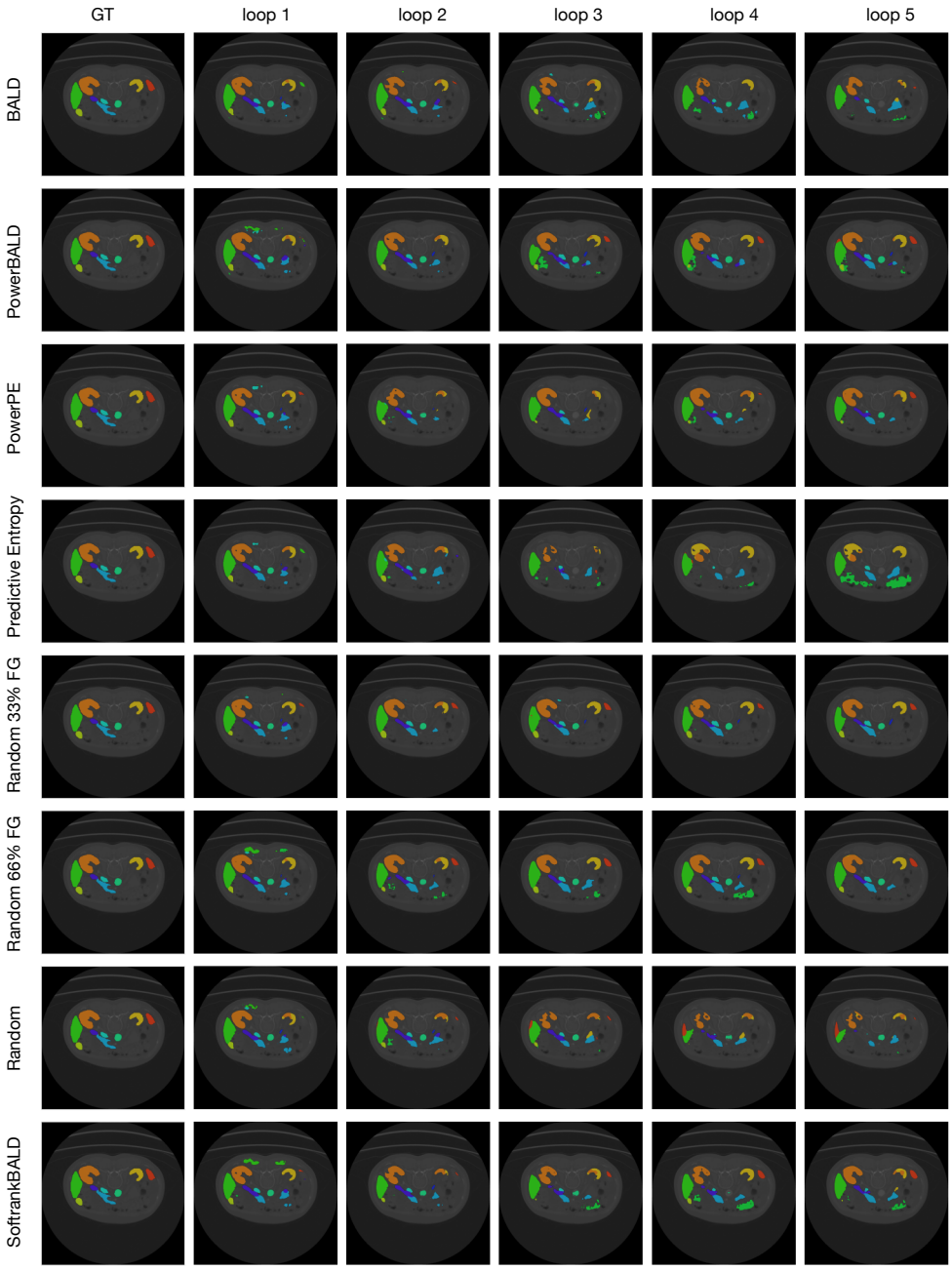


Figure C.15: **Exemplary Model Predictions on AMOS for different QMs.** Column 1: Ground Truth (GT) segmentation masks; Column 2-6: predicted segmentations after each AL loop.

Figure C.16: **Exemplary Model Predictions on Hippocampus for different QMs.** Column 1: Ground Truth (GT) segmentation masks; Column 2-6: predicted segmentations after each AL loop.

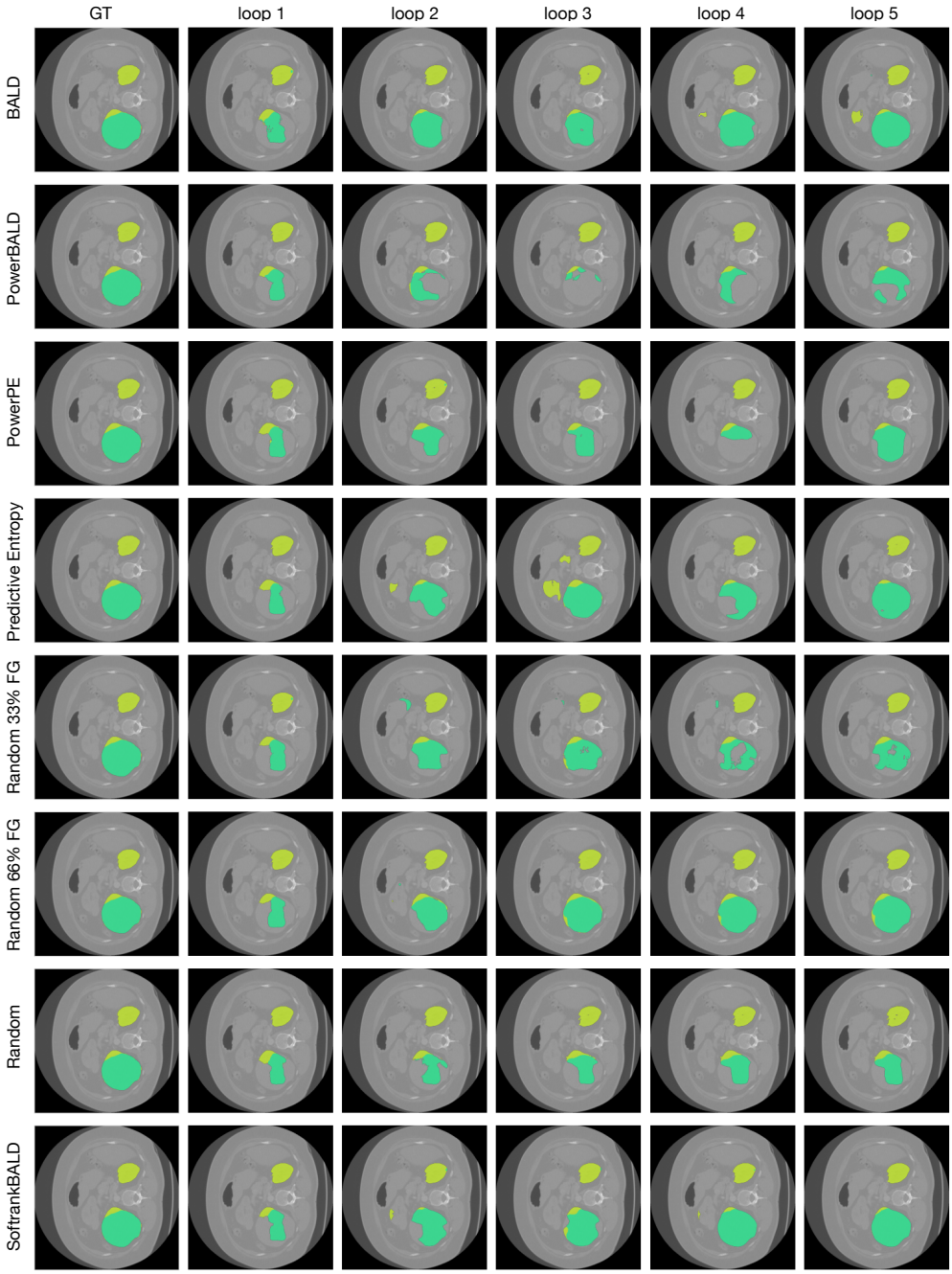


Figure C.17: **Exemplary Model Predictions on KiTS for different QMs.** Column 1: Ground Truth (GT) segmentation masks; Column 2-6: predicted segmentations after each AL loop.

Appendix D

A Simple Uncertainty-Based Active Learning Method for 3D Biomedical Segmentation

This chapter of the Appendix uses the Appendix of Carsten T. Lüth, Jeremias Traub, Kim-Celine Kahl, Till J. Bungert, Lukas Klein, Lars Krämer, Paul F. Jaeger, Klaus Maier-Hein, and Fabian Isensee (2025). "Finally outshining the Random Baseline: A simple and effective solution for Active Learning in 3D biomedical imaging". In: *Submitted to Transactions on Machine Learning Research*. Under review

D.1 ClaSP PE Algorithm

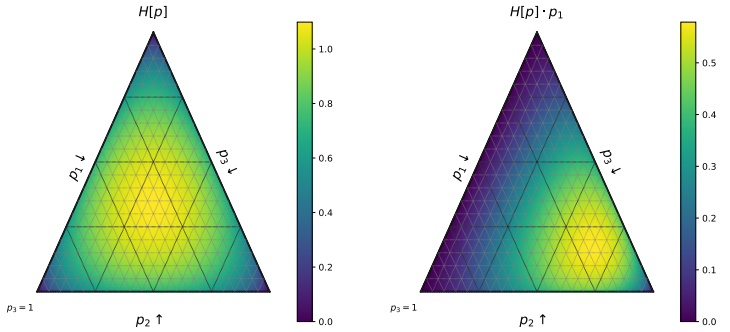


Figure D.1: Ternary plot visualizing the difference of the entropy $u = H[p]$ and our proposed class-specific measure $u_1 = H[p] \cdot p_1$ for $y \in \{1, 2, 3\}$.

We start by giving a short recap of our proposed query method (QM) to introduce the notation. Followed by additional implementation details to support reproducibility by means of two complementary representations of the algorithm for ClaSP PE.

Class Stratified Sampling Given an image x, an uncertainty map $u(x)$, and predicted class probabilities $p_c(x) = p(Y = c|x)$, we obtain the class-specific scores

$$u_c(x) = p_c(x) \cdot u(x) \tag{D.1}$$

A direct example of how these class specific scores behave in a class scenarios is visualized in fig. D.1. We then select samples in a stratified fashion for each class c based on u_c, respectively. To our

knowledge, this approach of balancing the queries using stratification has not been used in the AL literature before. Crucially, we do not select all samples with the stratified approach but only a fraction α with the remaining $1 - \alpha$ samples being selected based on the standard uncertainty map $u(x)$ to retain sensitivity to highly uncertain examples regardless of class distribution.

An Exponential Scheduler for Score Perturbation via Log-scale Power Noising Our exponential scheduled power-noising is a straight extension of the work by Kirsch, Farquhar, et al. (2023) works as follows:

$$s_{\text{ClaSP PE}}(t) = \log s_{\text{Cla PE}} + \epsilon(t) \tag{D.2}$$

where

$$\epsilon(t) \sim \text{Gumbel}(0, \beta^{-1}(t)) \tag{D.3}$$

with $t \in \{0, ..., T\}$ which represents the current AL cycle where T is the maximum number of AL cycles counting only those with a Query step. β_0 is the initial value, while $\beta_{\max}$ is the final value for the last cycle.

$$\beta(t) = \exp([1 - \tfrac{t}{T}] \ln(\beta_0) + \tfrac{t}{T} \ln(\beta_{\max})) \tag{D.4}$$

Implementations First, we provide a Python-style pseudocode in algorithm 2 that abstracts away specific implementation details, focusing instead on the core structure and logic of the method. Second, we present a fully detailed algorithmic version that outlines our exact implementation inside the nnActive framework shown in algorithm 3. This combination provides a high-level overview while also being transparent about our implementation.

As the high-level Python-style pseudocode abstracts away the patches, it therefore can serve as foundation for implementations where overlap checks are not necessary.

Algorithm 2 Abstracted ClaSP PE in a Python-style pseudocode with patches abstracted away

Input: unlabeled_pool: unlabeled dataset, model: python model, t: current loop, T: max loop with query, beta_0: starting beta, beta_max: final beta, alpha: fraction stratified, num_classes: number of classes, n: query size

PseudoCode

```
u_images = []
for x in unlabeled_pool: # Computing ClaSP PE for a sample
    p = model.forward(x)
    u = entropy(p)
    u_c = cat(p[without bg_class] * unsqueeze(u, 0), unsqueeze(u, 0))
    u_c += gumbel_noise(u_c.shape, exp(-(1-t/T)*ln(beta_0) + t/T *ln(beta_max))
    u_images.append(u_c)

# Selecting Query over entire samples s_budgets = floor(n*alpha/C)
query = [] for c in range(C[without bg_class]):
    best = argsort(u_images[:, c])
    best.pop(i) for i in query
    query.append(best[::-1][s_budgets])
best = argsort(u_images[:, c])
best.pop(i) for i in query
query.append(best[::-1][1- (s_budgets]*C))
return query
```

Algorithm 3 Exact ClaSP PE algorithm as implemented in the nnActive Framework

Input:
Set of images $\{X^{(i)}\}_{i=1}^{N}$, query size n, labeled set $\mathcal{L}$, Uncertainty function $\mathcal{U}$, number of classes C, fraction class specific α, aggregation method with scheduled powernoiseing (A)
Output: Final query set $\mathcal{Q}$

$\tilde{\mathcal{Q}} \leftarrow \{\emptyset\}_{c=1}^{C+1}$ # Initialize stratified query set
for each image $X^{(i)} \in \{X^{(i)}\}_{i=1}^{N}$ **do**
    $P \leftarrow \mathcal{M}(X)$ # compute probability for image
    $U \leftarrow U(X^{(i)}, \mathcal{M})$ # compute uncertainty for image
    $U_{\text{Agg}} \leftarrow A(\mathcal{U})$ # aggregate uncertainties to patch-level
    $Q_{\text{Image}} \leftarrow \{\emptyset\}_{c=1}^{C+1}$ # initialize best patches for current image
    for $c \in$ Shuffle($\{1, ..., C\}$) **do**
        $U_c \leftarrow U \cdot P_c$
        $U_{c,\text{Agg}} \leftarrow A(\mathcal{U})$ # aggregate uncertainties to patch-level
        for q in sort($U_{c,\text{Agg}}$)[::-1] **do** # sort in descending order according to uncertainty
            if overlap($q, \mathcal{Q}_{\text{Image}} \cup \mathcal{L}$) $\leq o$ **then** # ensure that
                $\mathcal{Q}_{c,\text{Image}} \leftarrow \mathcal{Q}_{c,\text{Image}} \cup \{q\}$
            end if
            if len($\mathcal{Q}_{c,\text{Image}}$) ¿= $\alpha * n/C$ **then**
                Break
            end if
        end for
    end for
    for q in sort(U_{Agg})[::-1] **do** # sort in descending order according to uncertainty
        if overlap($q, \mathcal{Q}_{\text{Image}} \cup \mathcal{L}$) $\leq o$ **then** # ensure that
            $\mathcal{Q}_{C+1,\text{Image}} \leftarrow \mathcal{Q}_{C+1,\text{Image}} \cup \{q\}$
        end if
        if len($\mathcal{Q}_{C+1,\text{Image}}$) ¿= $\alpha * n/C$ **then**
            Break
        end if
    end for
    $\tilde{\mathcal{Q}} \leftarrow \mathcal{Q} \cup \mathcal{Q}_{\text{Image}}$
end for
for $c \in \{1, ..., C\}$ # Build final query with stratified samples **do**
    $Q \leftarrow Q \cup \text{sort}(\tilde{Q}_c)[:: -1][: \alpha * n/C]$
end for
$Q \leftarrow Q \cup \text{sort}(\tilde{Q}_c)[:: -1][: n - (\alpha * n/C)]$ # Add unstratisfied samples
Return $\mathcal{Q}$

D.2 Dataset Details

Key characteristics for the roll-out study (section 8.3.2) are shown in table D.1. All images are resampled to the median dataset spacing. Further details on the different segmentation tasks are given in table D.2.

Table D.1: Dataset details and configurations for the roll-out study.

Dataset	LiTS	WORD	Tooth Fairy 2	MAMA MIA
# Classes w.o. Background	2	16	42	1
Median Shape	495×512×512	200×512×512	169×344×371	80×256×256
Used Spacing	1×0.7676×0.7676	3×0.9766×0.9766	0.3×0.3×0.3	2×0.7031×0.7031
# Pool & Training	99	90	360	1130
# Validation	32	30	120	376
Budget [# Patches](% Voxels)	750 (0.19%)	4,000 (15.8%)	10,500 (4.5%)	500 (0.09%)
Query Patch Size	28×44×39	29×74×87	33×34×35	16×48×57
Test set Mean Dice (1000 Epochs)	0.799	0.845	0.752	0.765
Test set Mean Dice (500 Epochs)	0.797	0.829	0.745	0.746
Test set Mean Dice (200 Epochs)	0.773	0.807	0.726	0.710

Table D.2: Foreground class names for all datasets.

Dataset	**Class names in order of labels (ascending)**
ACDC	right ventricle, myocardium, left ventricular cavity
AMOS	spleen, right kidney, left kidney, gall bladder, esophagus, liver, stomach, aorta, postcava, pancreas, right adrenal gland, left adrenal gland, duodenum, bladder, prostate/uterus
Hippocampus	anterior hippocampus, posterior hippocampus
KiTS	kidney, kidney-tumor, kidney-cyst
LiTS	liver, cancer
WORD	liver, spleen, left_kidney, right_kidney, stomach, gallbladder, esophagus, pancreas, duodenum, colon, intestine, adrenal, rectum, bladder, Head_of_femur_L, Head_of_femur_R
Tooth Fairy 2	Lower Jawbone, Upper Jawbone, Left Inferior Alveolar Canal, Right Inferior Alveolar Canal, Left Maxillary Sinus, Right Maxillary Sinus, Pharynx, Bridge, Crown, Implant, Upper Right Central Incisor, Upper Right Lateral Incisor, Upper Right Canine, Upper Right First Premolar, Upper Right Second Premolar, Upper Right First Molar, Upper Right Second Molar, Upper Right Third Molar (Wisdom Tooth), Upper Left Central Incisor, Upper Left Lateral Incisor, Upper Left Canine, Upper Left First Premolar, Upper Left Second Premolar, Upper Left First Molar, Upper Left Second Molar, Upper Left Third Molar (Wisdom Tooth), Lower Left Central Incisor, Lower Left Lateral Incisor, Lower Left Canine, Lower Left First Premolar, Lower Left Second Premolar, Lower Left First Molar, Lower Left Second Molar, Lower Left Third Molar (Wisdom Tooth), Lower Right Central Incisor, Lower Right Lateral Incisor, Lower Right Canine, Lower Right First Premolar, Lower Right Second Premolar, Lower Right First Molar, Lower Right Second Molar, Lower Right Third Molar (Wisdom Tooth)
MAMA MIA	lesion

D.3 Experiment Details

We follow the standard nnActive protocol for all Roll-Out experiments.

For the Tooth Fairy 2 dataset, we train without mirroring. For runtime savings, we omit Test-Time Augmentation during validation for MAMA MIA and Tooth Fairy 2.

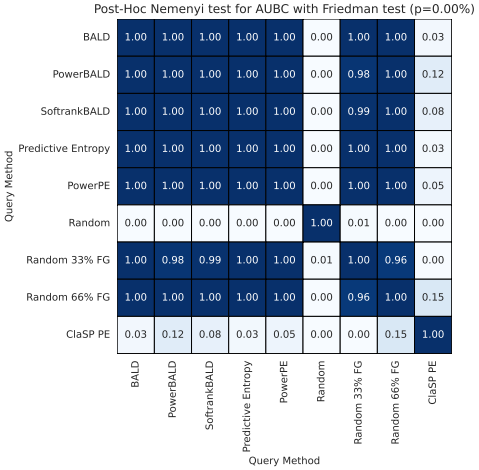
D.4 nnActive Benchmark Results

In this section, we provide detailed results on the nnActive benchmark. We refer to the *nnActive main benchmark* as the experiment configuration described in section 7.4.1, which encompasses 12 distinct settings across 4 datasets and 3 Label Regimes. Further extending the method evaluation, define a Patch$\times\frac{1}{2}$ setting, which uses a query patch size that is halved along each dimension compared to that of the main benchmark.

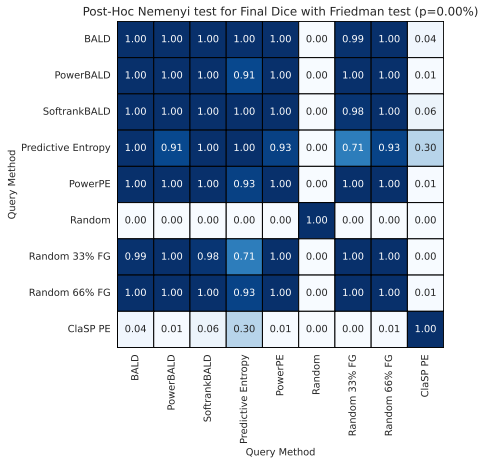
D.4.1 Results aggregated over Main Benchmark and Patch$\times\frac{1}{2}$ Setting

The results presented in this section are aggregated over both the main benchmark and the Patch$\times\frac{1}{2}$ setting, resulting in 24 distinct experiment configurations across 4 datasets, 3 Label Regimes, and 2 query patch sizes. Specifically, fig. D.2 shows the results of Nemenyi post-hoc tests, based on Friedman tests (Demšar, 2006), to analyze the significance of performance differences, and fig. D.3 shows the PPMs for each dataset.

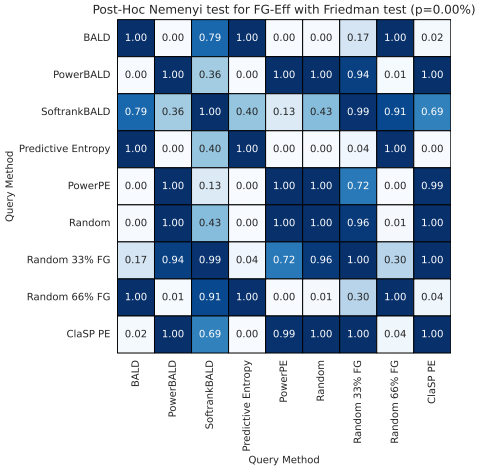
For the Nemenyi post-hoc tests, we selected $p = 0.05$ as the nature of the test is very conservative (Nemenyi, 1963) and since our sample sizes are comparatively low (especially considering the number of methods we compare). Moreover, all methods have the same chance of outperforming the random baselines.



(a) AUBC

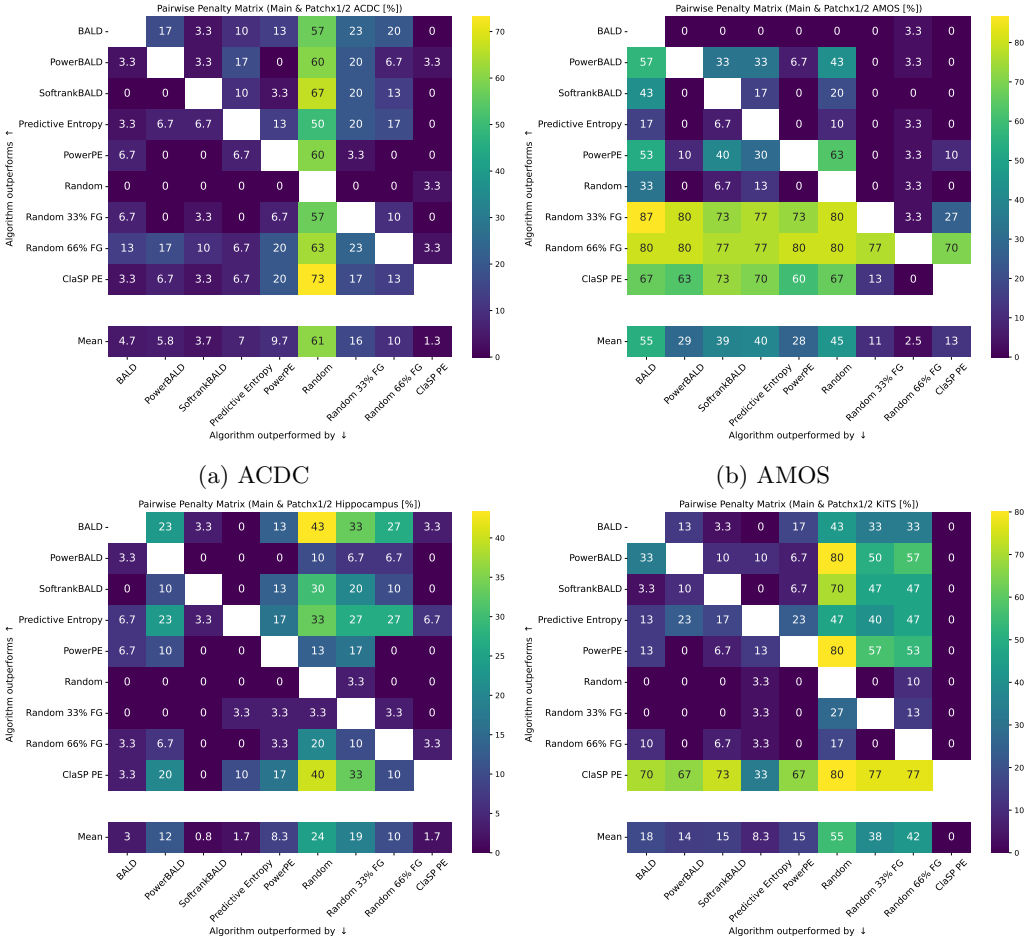


(b) Final Dice



(c) FG-Eff

Figure D.2: p-values for the Nemenyi post-hoc tests, based on Friedman tests, on the nnActive benchmark for all evaluation metrics. Results are aggregated across 4 datasets $\times$ 3 Label Regimes $\times$ 2 query patch sizes. The corresponding significance groups for $p = 0.05$ are indicated in fig. 8.2.



(a) ACDC

(b) AMOS

(c) Hippocampus

(d) KiTS

Figure D.3: Pairwise Penalty Matrices aggregated over all Label Regimes and both query patch sizes for each dataset.

D.4.2 Main Benchmark Results

The results shown in this section are obtained on the nnActive main study settings. Detailed results for AUBC, Final Dice, and FG-Eff, including standard deviations based on four seeds, are provided in table D.3. The table includes results for the methods Cla PE 66% and 33%, as assessed in section 8.3.1. The overall PPM is shown in fig. D.4, the respective dataset-specific PPMs are in fig. D.5.

Table D.3: Fine-grained Results for the nnActive Main Study for each dataset. Higher values are better, and colorization goes from dark green (best) to white (worst) with linear interpolation. AUBC and Final Dice are multiplied ×100 for improved readability. AUBC, Final, and FG-Eff can only be directly compared within each Label Regime on each dataset.

(a) ACDC

Dataset	ACDC								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	79.84 ± 0.59	86.44 ± 0.96	26.98 ± 3.11	85.85 ± 0.45	89.62 ± 0.15	21.91 ± 4.20	87.74 ± 0.38	90.47 ± 0.18	15.09 ± 1.14
PowerBALD	81.18 ± 0.58	86.46 ± 0.55	46.29 ± 13.10	85.63 ± 0.37	89.07 ± 0.21	27.75 ± 4.00	87.50 ± 0.44	89.80 ± 0.17	17.94 ± 1.83
SoftrankBALD	80.71 ± 0.92	86.50 ± 0.95	35.71 ± 7.09	85.89 ± 0.49	89.33 ± 0.27	26.33 ± 5.01	87.28 ± 0.68	90.17 ± 0.14	14.53 ± 1.33
Predictive Entropy	80.02 ± 1.54	86.54 ± 0.95	26.49 ± 4.40	85.53 ± 0.59	89.42 ± 0.07	21.16 ± 3.11	87.65 ± 0.27	90.52 ± 0.06	13.58 ± 1.22
PowerPE	80.46 ± 0.30	86.56 ± 0.40	47.88 ± 14.09	85.24 ± 0.69	89.05 ± 0.22	27.92 ± 5.01	87.21 ± 0.60	89.67 ± 0.15	16.55 ± 1.18
Random	76.65 ± 0.81	80.34 ± 1.64	59.25 ± 33.53	82.24 ± 1.25	83.46 ± 0.87	38.22 ± 8.43	84.69 ± 0.96	86.28 ± 1.08	21.69 ± 3.79
Random 33% FG	81.28 ± 0.56	85.09 ± 1.14	40.88 ± 9.71	84.61 ± 0.65	87.51 ± 0.56	21.26 ± 1.49	86.95 ± 0.74	89.06 ± 0.44	15.81 ± 1.41
Random 66% FG	82.32 ± 0.33	86.70 ± 0.48	31.20 ± 4.32	86.16 ± 0.44	88.62 ± 0.52	18.95 ± 2.13	87.86 ± 0.33	89.94 ± 0.09	13.44 ± 0.79
Cla PE 33%	81.00 ± 0.74	86.38 ± 0.84	28.70 ± 2.71	85.67 ± 0.55	89.57 ± 0.09	19.93 ± 2.28	87.83 ± 0.37	90.50 ± 0.20	14.04 ± 0.92
Cla PE 66%	82.12 ± 0.71	87.45 ± 0.87	28.30 ± 2.47	86.11 ± 0.23	89.66 ± 0.15	18.04 ± 1.39	88.05 ± 0.15	90.55 ± 0.06	13.86 ± 1.00
ClaP PE	80.40 ± 0.55	86.11 ± 0.50	39.52 ± 8.07	86.33 ± 0.67	89.27 ± 0.47	33.61 ± 6.39	87.67 ± 0.35	89.97 ± 0.12	19.77 ± 2.01
ClaSP PE	81.31 ± 0.47	86.88 ± 0.78	37.36 ± 7.41	86.44 ± 0.67	89.50 ± 0.31	31.97 ± 11.86	87.91 ± 0.36	90.56 ± 0.09	18.66 ± 3.43

(b) AMOS

Dataset	AMOS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	38.69 ± 2.34	34.05 ± 1.58	-22.65 ± 8.50	52.56 ± 2.74	59.26 ± 2.73	1.49 ± 0.22	69.38 ± 0.70	74.95 ± 2.38	-0.45 ± 0.20
PowerBALD	50.34 ± 3.00	56.18 ± 1.24	3.67 ± 14.54	66.11 ± 1.47	73.02 ± 2.01	18.19 ± 0.44	77.86 ± 0.14	80.48 ± 0.48	8.78 ± 0.08
SoftrankBALD	44.49 ± 1.56	45.75 ± 0.95	-11.37 ± 4.19	60.01 ± 0.69	66.72 ± 0.65	5.66 ± 0.10	75.29 ± 1.46	81.23 ± 1.18	3.51 ± 0.39
Predictive Entropy	38.02 ± 3.35	39.19 ± 6.79	-17.91 ± 8.48	56.30 ± 1.78	62.07 ± 1.39	2.62 ± 0.17	71.27 ± 1.52	80.79 ± 2.07	1.01 ± 0.41
PowerPE	47.66 ± 2.50	50.04 ± 2.30	-9.78 ± 12.12	66.74 ± 2.80	73.68 ± 0.92	18.51 ± 1.17	77.92 ± 0.29	80.52 ± 0.16	8.86 ± 0.10
Random	42.26 ± 2.55	36.36 ± 2.92	-134.74 ± 88.92	54.65 ± 2.82	56.22 ± 4.61	10.09 ± 3.26	73.82 ± 0.50	75.48 ± 0.37	7.33 ± 0.62
Random 33% FG	58.05 ± 1.54	62.95 ± 1.03	35.47 ± 11.41	71.78 ± 1.16	78.60 ± 0.37	36.44 ± 2.94	79.53 ± 0.38	82.68 ± 0.19	14.42 ± 0.47
Random 66% FG	62.84 ± 1.88	71.11 ± 1.42	43.64 ± 9.81	74.87 ± 0.64	80.72 ± 0.54	32.50 ± 6.08	80.98 ± 0.19	83.81 ± 0.32	12.32 ± 0.43
Cla PE 33%	45.98 ± 2.14	49.85 ± 1.01	-6.04 ± 3.50	64.20 ± 2.09	71.54 ± 3.62	6.62 ± 0.21	79.52 ± 0.49	83.57 ± 0.39	5.96 ± 0.04
Cla PE 66%	51.66 ± 1.49	53.35 ± 1.75	1.00 ± 1.10	68.90 ± 1.71	78.50 ± 0.92	10.22 ± 0.26	80.84 ± 0.18	84.70 ± 0.07	7.47 ± 0.04
ClaP PE	53.60 ± 2.03	59.60 ± 3.92	15.86 ± 13.73	70.61 ± 1.45	78.51 ± 0.52	25.17 ± 0.97	79.83 ± 0.24	83.22 ± 0.26	11.34 ± 0.27
ClaSP PE	54.15 ± 2.26	59.82 ± 4.15	11.56 ± 6.25	71.28 ± 1.23	79.54 ± 0.29	20.01 ± 2.24	80.63 ± 0.12	84.40 ± 0.18	10.62 ± 0.60

(c) Hippocampus

Dataset	Hippocampus								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	88.46 ± 0.03	88.87 ± 0.06	9.58 ± 0.98	88.79 ± 0.02	89.18 ± 0.07	4.52 ± 0.06	89.03 ± 0.05	89.42 ± 0.05	3.49 ± 0.12
PowerBALD	88.20 ± 0.08	88.77 ± 0.11	9.21 ± 0.49	88.76 ± 0.04	89.16 ± 0.06	5.56 ± 0.07	88.98 ± 0.07	89.29 ± 0.10	3.90 ± 0.15
SoftrankBALD	88.44 ± 0.11	88.93 ± 0.18	9.61 ± 0.98	88.72 ± 0.08	89.12 ± 0.02	3.90 ± 0.05	89.03 ± 0.06	89.42 ± 0.07	3.60 ± 0.12
Predictive Entropy	88.50 ± 0.06	88.90 ± 0.10	9.75 ± 1.01	88.81 ± 0.04	89.18 ± 0.07	4.23 ± 0.06	89.07 ± 0.07	89.54 ± 0.03	3.73 ± 0.19
PowerPE	88.16 ± 0.08	88.70 ± 0.11	9.25 ± 0.52	88.63 ± 0.09	89.07 ± 0.21	4.41 ± 0.10	88.97 ± 0.07	89.33 ± 0.18	4.08 ± 0.24
Random	88.07 ± 0.10	88.58 ± 0.08	8.76 ± 0.47	88.65 ± 0.11	89.07 ± 0.04	5.10 ± 0.08	88.96 ± 0.09	89.29 ± 0.20	4.41 ± 0.25
Random 33% FG	88.22 ± 0.16	88.70 ± 0.08	9.60 ± 0.81	88.77 ± 0.13	89.22 ± 0.14	6.21 ± 0.17	88.94 ± 0.06	89.33 ± 0.10	3.85 ± 0.15
Random 66% FG	88.28 ± 0.13	88.76 ± 0.14	9.88 ± 0.73	88.63 ± 0.02	89.02 ± 0.04	4.21 ± 0.03	88.92 ± 0.08	89.26 ± 0.06	3.33 ± 0.11
Cla PE 33%	88.49 ± 0.06	88.97 ± 0.20	9.73 ± 0.94	88.88 ± 0.05	89.22 ± 0.08	5.21 ± 0.08	89.04 ± 0.05	89.43 ± 0.00	3.48 ± 0.21
Cla PE 66%	88.43 ± 0.10	88.90 ± 0.14	8.99 ± 0.64	88.77 ± 0.03	89.08 ± 0.12	4.02 ± 0.08	89.03 ± 0.06	89.46 ± 0.08	3.51 ± 0.14
ClaP PE	88.21 ± 0.13	88.64 ± 0.14	9.27 ± 0.71	88.69 ± 0.06	89.11 ± 0.08	5.28 ± 0.08	88.91 ± 0.07	89.25 ± 0.06	3.36 ± 0.11
ClaSP PE	88.28 ± 0.12	88.89 ± 0.13	9.59 ± 0.71	88.70 ± 0.11	89.15 ± 0.14	4.79 ± 0.11	88.97 ± 0.11	89.41 ± 0.09	3.86 ± 0.22

(d) KiTS

Dataset	KiTS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	40.58 ± 2.75	44.03 ± 3.18	7.96 ± 0.82	55.06 ± 1.20	61.97 ± 1.49	6.51 ± 0.14	62.53 ± 0.84	67.57 ± 1.72	9.37 ± 0.46
PowerBALD	45.10 ± 2.91	47.67 ± 3.63	25.24 ± 6.06	54.53 ± 1.40	59.51 ± 1.15	10.16 ± 0.41	61.24 ± 0.57	65.04 ± 0.81	11.92 ± 0.64
SoftrankBALD	42.87 ± 2.91	47.12 ± 3.34	12.41 ± 2.03	54.83 ± 1.79	61.44 ± 2.02	6.99 ± 0.27	62.49 ± 0.74	67.00 ± 0.97	9.84 ± 0.66
Predictive Entropy	40.62 ± 2.74	45.53 ± 3.57	7.05 ± 0.64	57.42 ± 0.54	65.39 ± 0.51	6.19 ± 0.10	64.00 ± 0.15	68.74 ± 0.65	7.84 ± 0.21
PowerPE	45.30 ± 2.05	49.62 ± 1.13	28.70 ± 3.74	54.76 ± 1.10	58.67 ± 1.53	9.68 ± 0.28	60.66 ± 0.66	63.62 ± 1.19	9.62 ± 0.51
Random	38.75 ± 3.36	39.19 ± 4.13	28.47 ± 19.48	47.82 ± 1.84	48.41 ± 1.99	4.03 ± 2.75	53.80 ± 0.68	55.12 ± 1.27	8.93 ± 1.22
Random 33% FG	43.70 ± 0.87	47.35 ± 2.10	16.19 ± 1.33	51.50 ± 1.97	54.08 ± 2.76	3.27 ± 0.15	55.30 ± 1.26	56.79 ± 1.02	1.88 ± 0.04
Random 66% FG	44.97 ± 2.01	46.83 ± 2.53	11.28 ± 1.30	50.78 ± 0.97	51.67 ± 2.31	1.24 ± 0.02	53.73 ± 1.78	55.90 ± 0.84	0.68 ± 0.01
Cla PE 33%	45.62 ± 2.32	53.07 ± 1.36	12.70 ± 0.60	59.63 ± 0.73	66.41 ± 0.98	7.51 ± 0.07	64.82 ± 0.42	69.09 ± 0.49	8.73 ± 0.30
Cla PE 66%	48.09 ± 2.00	54.30 ± 2.46	13.97 ± 0.65	61.27 ± 0.63	68.42 ± 0.46	8.08 ± 0.09	65.58 ± 0.62	69.60 ± 0.35	8.70 ± 0.23
ClaP PE	46.80 ± 1.96	52.72 ± 1.65	29.08 ± 3.47	59.22 ± 1.46	63.91 ± 1.21	12.82 ± 0.75	63.74 ± 0.28	67.68 ± 0.85	11.66 ± 0.71
ClaSP PE	47.77 ± 1.63	54.83 ± 1.70	18.49 ± 2.11	60.33 ± 0.87	66.97 ± 0.91	10.38 ± 0.70	64.50 ± 0.29	69.53 ± 0.68	11.20 ± 1.25

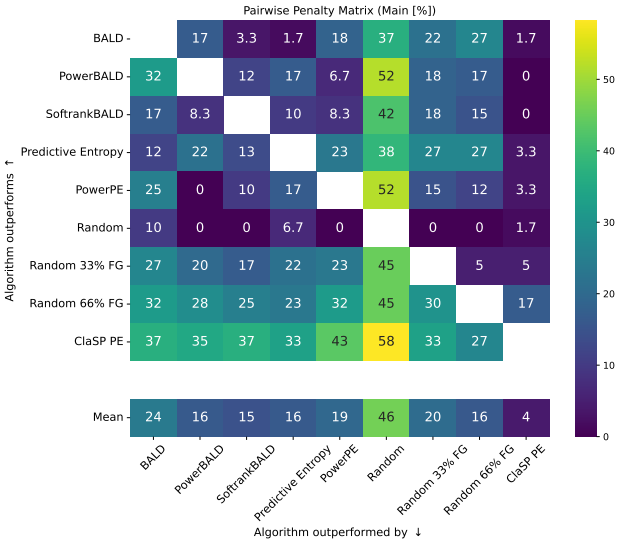
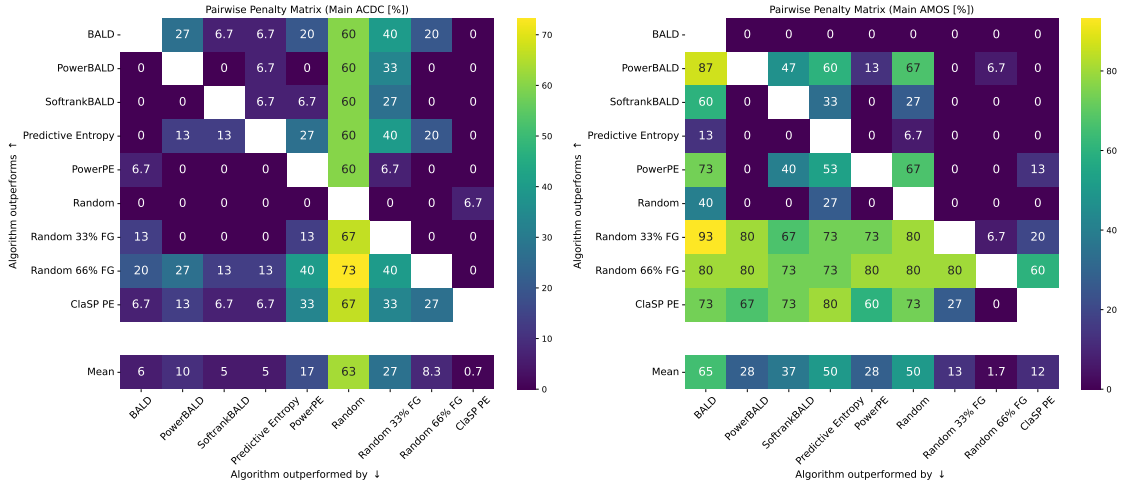


Figure D.4: PPM aggregated over the nnActive main study experiments.



(a) ACDC

(b) AMOS

(c) Hippocampus

(d) KiTS

Figure D.5: Pairwise Penalty Matrix aggregated over all Label Regimes for each dataset of the main study.

D.4.3 Patch$\times\frac{1}{2}$ Setting results

Analogous to section D.4.2, this section provides results for the Patch$\times\frac{1}{2}$ settings, including a detailed results table (table D.4), and the overall (fig. D.6) and datset-specific (fig. D.7) PPMs.

Table D.4: Fine-grained Results for the patch ablation with setting Patch$\times\frac{1}{2}$ for each dataset. Higher values are better, and colorization goes from dark green (best) to white (worst) with linear interpolation. AUBC and Final Dice are multiplied $\times 100$ for improved readability. AUBC, Final, and FG-Eff can only be directly compared for each Label Regime on each dataset.

(a) ACDC

Dataset	ACDC								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	68.23 ± 2.31	77.72 ± 1.46	230.38 ± 440.89	75.80 ± 1.10	82.59 ± 1.24	149.81 ± 438.18	79.59 ± 1.05	83.99 ± 0.85	75.49 ± 69.33
PowerBALD	65.90 ± 5.61	75.24 ± 2.32	305.90 ± 942.82	77.07 ± 1.11	83.01 ± 1.21	207.99 ± 497.01	80.27 ± 1.39	84.54 ± 1.25	121.51 ± 193.59
SoftrankBALD	66.81 ± 3.68	75.98 ± 0.13	245.20 ± 442.21	76.84 ± 1.31	82.16 ± 0.47	199.23 ± 461.60	79.69 ± 1.03	84.03 ± 1.56	118.32 ± 181.98
Predictive Entropy	65.27 ± 2.45	75.79 ± 2.45	184.91 ± 201.36	74.67 ± 1.26	81.18 ± 1.32	119.52 ± 162.65	79.25 ± 0.95	83.58 ± 1.33	84.96 ± 102.80
PowerPE	65.70 ± 3.90	74.46 ± 2.28	300.97 ± 888.93	76.26 ± 2.36	82.16 ± 2.15	211.91 ± 452.95	79.85 ± 1.31	84.48 ± 1.55	132.75 ± 269.80
Random	59.38 ± 5.56	65.19 ± 4.17	479.15 ± 2311.84	70.99 ± 3.17	76.66 ± 1.22	461.53 ± 714.23	76.30 ± 0.80	79.09 ± 0.46	260.93 ± 529.78
Random 33% FG	67.98 ± 1.51	77.43 ± 0.27	216.14 ± 96.27	75.09 ± 1.78	81.65 ± 1.01	127.87 ± 65.39	79.55 ± 1.12	84.44 ± 0.32	87.88 ± 27.50
Random 66% FG	64.33 ± 1.17	73.97 ± 0.55	101.64 ± 46.52	74.69 ± 0.28	82.18 ± 1.52	72.09 ± 21.05	80.33 ± 0.56	85.88 ± 0.64	56.90 ± 8.42
ClaSP PE	68.00 ± 4.45	76.43 ± 2.90	239.84 ± 935.98	75.99 ± 2.64	82.60 ± 1.68	143.57 ± 329.15	79.82 ± 1.09	84.24 ± 0.79	88.31 ± 136.97

(b) AMOS

Dataset	AMOS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	13.98 ± 1.24	10.96 ± 2.19	-150.74 ± 374.12	16.93 ± 2.33	17.85 ± 4.60	-10.09 ± 20.38	30.15 ± 1.72	27.72 ± 0.83	-19.23 ± 3.35
PowerBALD	14.54 ± 2.70	11.74 ± 2.59	-248.75 ± 772.72	21.71 ± 1.48	25.83 ± 2.25	9.30 ± 15.98	40.14 ± 1.86	42.40 ± 1.47	8.53 ± 5.05
SoftrankBALD	13.63 ± 2.69	11.39 ± 1.68	-127.75 ± 359.40	19.95 ± 1.55	23.48 ± 3.19	-0.60 ± 8.12	35.13 ± 2.40	39.37 ± 1.87	-3.05 ± 5.38
Predictive Entropy	13.83 ± 2.12	12.28 ± 1.98	-83.91 ± 260.52	24.61 ± 2.34	27.37 ± 5.21	7.21 ± 2.16	36.63 ± 6.19	43.86 ± 6.88	1.71 ± 4.83
PowerPE	15.18 ± 2.85	13.00 ± 4.65	-212.38 ± 565.02	23.28 ± 1.26	27.05 ± 2.03	15.28 ± 8.68	43.20 ± 1.78	47.34 ± 3.06	18.72 ± 3.42
Random	12.78 ± 2.02	8.89 ± 1.91	-942.96 ± 5829.09	16.14 ± 1.62	16.99 ± 3.32	-89.28 ± 165.24	37.56 ± 1.57	37.28 ± 3.44	-3.38 ± 23.05
Random 33% FG	22.10 ± 1.18	24.14 ± 4.37	14.60 ± 105.81	39.32 ± 2.68	51.61 ± 4.21	103.81 ± 24.46	56.68 ± 1.86	65.54 ± 1.82	63.64 ± 11.01
Random 66% FG	31.10 ± 2.19	39.70 ± 0.34	134.71 ± 67.16	48.12 ± 0.68	60.25 ± 0.45	94.92 ± 28.62	62.07 ± 0.73	70.71 ± 0.60	51.62 ± 8.54
ClaSP PE	20.32 ± 2.58	25.84 ± 2.35	6.94 ± 99.73	31.54 ± 2.77	43.85 ± 2.68	36.96 ± 8.94	53.15 ± 3.14	67.43 ± 1.35	32.86 ± 3.25

(c) Hippocampus

Dataset	Hippocampus								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	86.42 ± 0.47	87.85 ± 0.15	72.18 ± 173.53	87.64 ± 0.17	88.43 ± 0.19	15.02 ± 2.49	87.99 ± 0.16	88.76 ± 0.08	12.46 ± 1.33
PowerBALD	86.07 ± 0.35	87.45 ± 0.37	79.12 ± 97.84	87.32 ± 0.04	88.12 ± 0.04	18.16 ± 1.28	87.82 ± 0.04	88.47 ± 0.07	17.28 ± 1.29
SoftrankBALD	86.44 ± 0.33	87.66 ± 0.32	73.07 ± 153.84	87.54 ± 0.17	88.27 ± 0.10	16.63 ± 2.19	87.92 ± 0.07	88.66 ± 0.14	15.13 ± 1.68
Predictive Entropy	86.34 ± 0.22	87.69 ± 0.09	63.64 ± 128.00	87.43 ± 0.14	88.41 ± 0.09	13.37 ± 1.22	87.99 ± 0.14	88.74 ± 0.09	12.30 ± 1.71
PowerPE	86.21 ± 0.70	87.56 ± 0.51	84.35 ± 144.15	87.43 ± 0.11	88.29 ± 0.11	19.81 ± 2.65	87.94 ± 0.11	88.43 ± 0.15	18.11 ± 2.41
Random	85.62 ± 0.65	86.74 ± 0.31	118.43 ± 222.57	87.06 ± 0.21	87.76 ± 0.10	25.64 ± 5.49	87.58 ± 0.15	88.13 ± 0.15	26.07 ± 3.15
Random 33% FG	85.69 ± 0.56	87.02 ± 0.19	78.79 ± 124.77	87.26 ± 0.17	88.00 ± 0.07	17.19 ± 2.51	87.74 ± 0.06	88.31 ± 0.11	15.41 ± 1.27
Random 66% FG	86.24 ± 0.13	87.54 ± 0.15	57.16 ± 55.78	87.49 ± 0.21	88.27 ± 0.10	14.91 ± 1.15	87.85 ± 0.21	88.54 ± 0.17	12.43 ± 0.64
ClaSP PE	86.62 ± 0.41	87.80 ± 0.28	79.67 ± 214.16	87.66 ± 0.06	88.43 ± 0.05	14.89 ± 2.08	87.96 ± 0.10	88.59 ± 0.08	12.64 ± 1.97

(d) KiTS

Dataset	KiTS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	25.10 ± 0.55	31.76 ± 4.51	86.79 ± 97.07	38.56 ± 3.27	43.25 ± 3.79	31.30 ± 10.14	48.46 ± 1.19	53.50 ± 1.28	23.96 ± 6.78
PowerBALD	27.91 ± 1.74	29.39 ± 1.30	184.72 ± 578.47	41.70 ± 1.09	45.59 ± 1.40	78.84 ± 44.77	49.60 ± 0.95	54.00 ± 1.25	43.02 ± 17.06
SoftrankBALD	25.67 ± 2.68	31.47 ± 1.54	90.70 ± 90.22	41.08 ± 1.00	46.14 ± 1.77	46.46 ± 16.84	49.08 ± 1.12	54.39 ± 1.53	31.73 ± 10.26
Predictive Entropy	24.08 ± 1.56	29.07 ± 5.82	41.77 ± 25.38	40.99 ± 3.00	46.80 ± 3.63	23.97 ± 2.94	50.22 ± 1.42	55.79 ± 1.07	14.86 ± 1.67
PowerPE	27.96 ± 3.53	30.88 ± 4.84	207.47 ± 651.68	42.26 ± 0.77	46.55 ± 0.95	83.36 ± 51.36	49.48 ± 1.57	53.59 ± 1.03	44.14 ± 21.87
Random	22.00 ± 1.62	22.85 ± 2.15	139.89 ± 529.13	35.14 ± 2.00	37.95 ± 1.74	93.75 ± 139.60	42.73 ± 1.09	44.35 ± 1.60	46.87 ± 73.85
Random 33% FG	23.88 ± 3.43	28.83 ± 1.46	49.00 ± 31.48	37.88 ± 0.74	41.24 ± 0.88	17.55 ± 1.47	42.28 ± 1.19	44.19 ± 1.31	3.48 ± 0.15
Random 66% FG	24.43 ± 1.96	28.80 ± 2.90	29.82 ± 7.76	34.12 ± 1.40	36.52 ± 1.24	4.34 ± 0.27	40.24 ± 1.31	42.58 ± 0.88	0.68 ± 0.04
ClaSP PE	32.16 ± 1.04	40.16 ± 0.63	98.74 ± 133.82	45.76 ± 0.53	52.66 ± 0.97	31.13 ± 15.63	53.69 ± 0.93	59.96 ± 2.14	19.75 ± 5.40

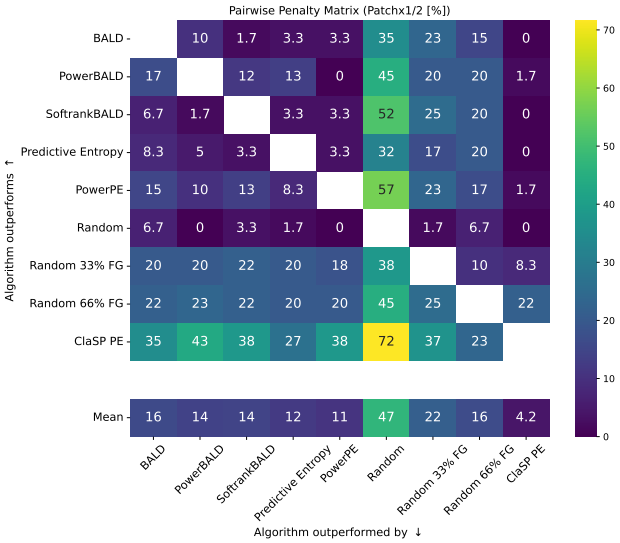
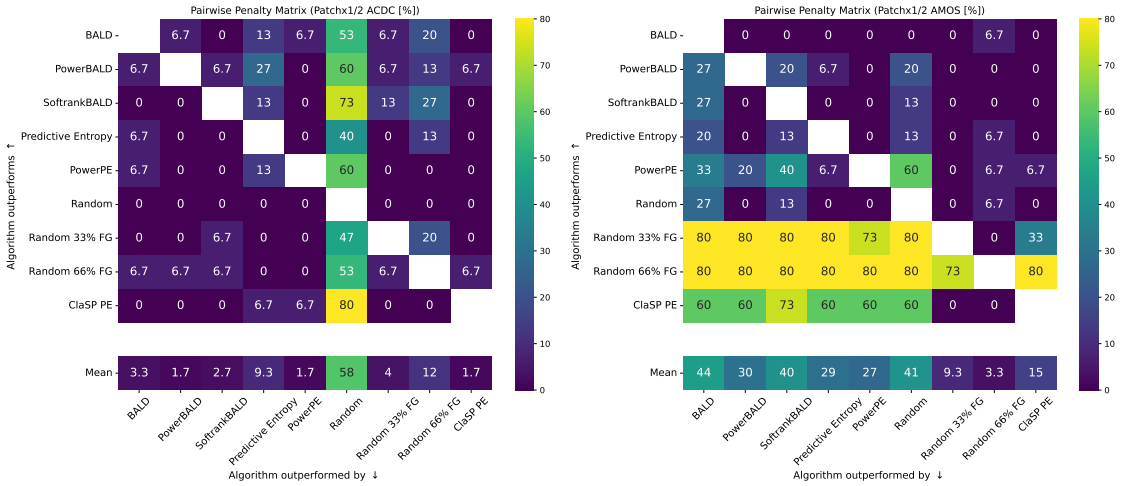


Figure D.6: PPM aggregated over the Patch$\times\frac{1}{2}$ settings. Mean row results change compared to the nnActive main study (fig. D.4).



(a) ACDC

(b) AMOS

(c) Hippocampus

(d) KiTS

Figure D.7: Pairwise Penalty Matrix aggregated over all Label Regimes for each dataset for the Patch$\times\frac{1}{2}$ ablation.

D.4.4 500 Epochs Setting results

For the ablation on the loss scenarios on AMOS in section 8.3.1, we provide detailed results in table D.5.

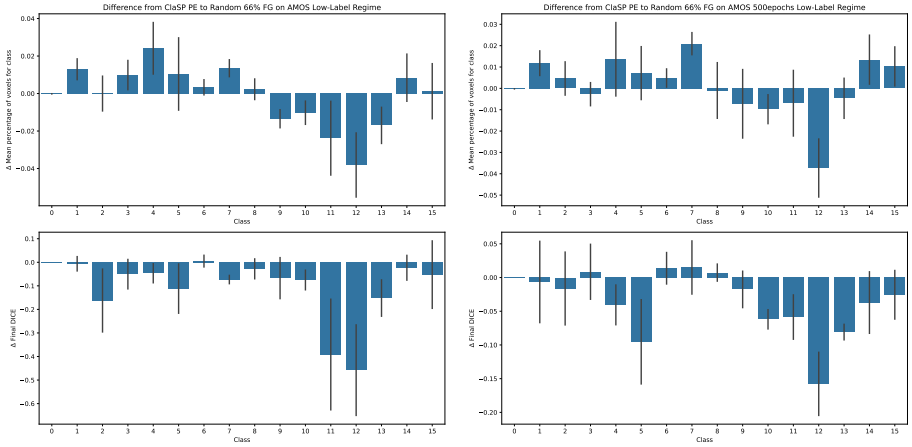
Table D.5: Fine-grained Results for the AMOS experiments when training for 500 epochs. Higher values are better, and colorization goes from dark green (best) to white (worst) with linear interpolation. AUBC and Final Dice are multiplied ×100 for improved readability. AUBC, Final, and FG-Eff can only be directly compared for each Label Regime on each dataset.

Dataset	AMOS								
Label Regime	Low			Medium			High		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method									
BALD	–	–	–	75.76 ± 1.20	81.67 ± 2.40	7.90 ± 0.54	84.37 ± 0.10	87.18 ± 0.07	7.71 ± 0.12
PowerBALD	–	–	–	79.87 ± 0.33	83.96 ± 0.41	24.98 ± 1.10	84.50 ± 0.16	86.35 ± 0.04	12.57 ± 0.26
SoftrankBALD	–	–	–	79.04 ± 0.29	83.55 ± 0.08	14.54 ± 0.33	84.60 ± 0.26	86.83 ± 0.16	9.27 ± 0.11
Predictive Entropy	–	–	–	77.21 ± 0.53	83.40 ± 0.41	9.19 ± 0.25	84.70 ± 0.03	87.52 ± 0.08	7.09 ± 0.09
PowerPE	–	–	–	79.27 ± 0.36	83.35 ± 0.21	22.80 ± 0.69	84.32 ± 0.32	86.23 ± 0.19	11.59 ± 0.27
Random	–	–	–	–	–	–	–	–	–
Random 33% FG	65.08 ± 1.59	71.22 ± 2.39	64.00 ± 23.13	79.41 ± 0.48	83.40 ± 0.35	33.90 ± 3.19	84.16 ± 0.14	86.04 ± 0.13	14.43 ± 0.49
Random 66% FG	68.76 ± 1.38	77.13 ± 0.55	60.89 ± 13.80	81.28 ± 0.51	85.11 ± 0.30	27.99 ± 2.92	85.10 ± 0.24	86.96 ± 0.24	12.61 ± 0.46
Cla PE 66%	65.18 ± 0.80	71.87 ± 0.85	24.29 ± 2.84	80.37 ± 0.56	85.23 ± 0.30	14.08 ± 0.21	85.38 ± 0.02	87.66 ± 0.11	8.69 ± 0.07
ClaSP PE	64.73 ± 0.30	73.44 ± 1.30	36.02 ± 6.94	80.74 ± 0.48	85.17 ± 0.24	23.55 ± 3.92	85.37 ± 0.15	87.63 ± 0.08	13.73 ± 0.96

D.4.5 Analyzing ClaSP PE performance on AMOS on a class level

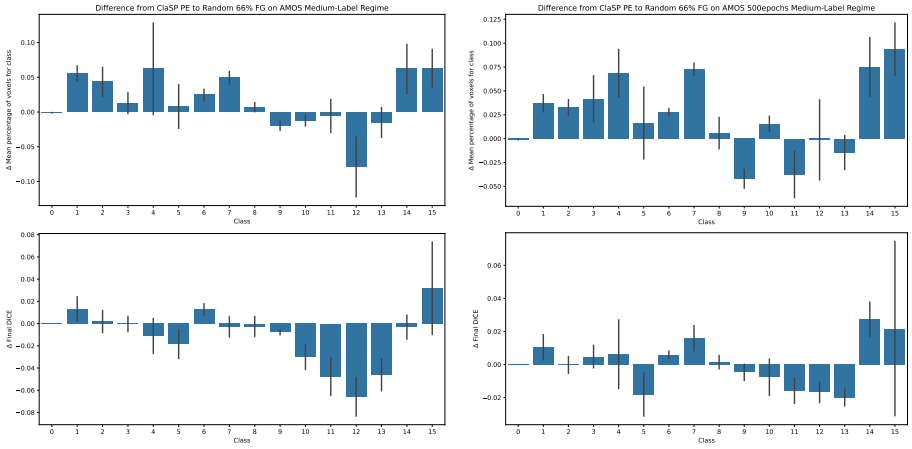
We analyze the performance of ClaSP PE relative to Random 66%FG on the Final Dice with regard to each class and the percentage of voxels queried on the AMOS dataset for 200 and 500 epochs on the main setting across all Label Regimes in fig. D.8. We observe that the longer training leads to ClaSP PE gaining more performance than Random 66% FG, but it also leads to a general increase in queried foreground. Overall, the esophagus, postcava, pancreas, right adrenal gland, left adrenal gland and duodenum (classes 5, 9, 10, 11, 12 and 13) are the most challenging classes. Importantly, the performance differences for right and left adrenal gland reduce with larger annotation budgets and longer training. While the largest performance gains stem from the bladder and prostate (classes 14 and 15).

In the low-Label Regime, especially the classes 11 and 12 (right and left adrenal gland), lead to relative performance losses of ClaSP PE relative to Random 66% FG which are also less frequently queried. In the medium and low-Label Regime with more queries for these classes this performance difference diminishes for 200 epochs and vanishes for 500 epochs below random noise.



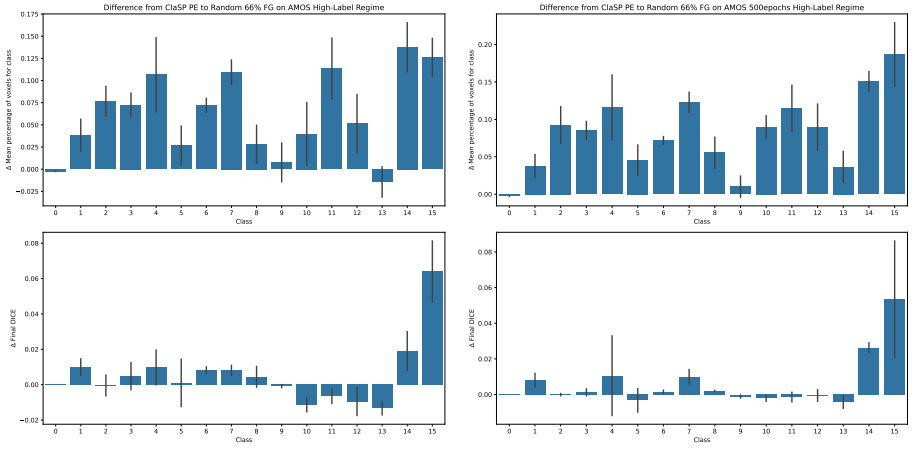
(a) Low-Label - 200 Epochs

(b) Low-Label - 500 Epochs



(c) Medium-Label - 200 Epochs

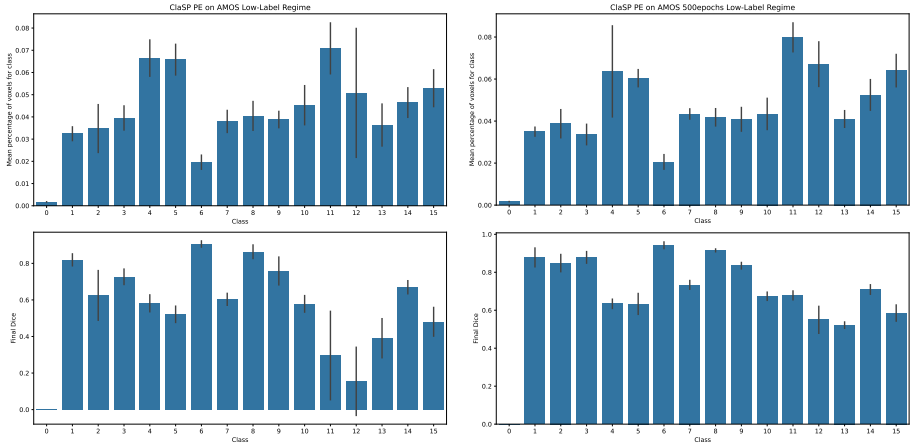
(d) Medium-Label - 500 Epochs



(e) High-Label - 200 Epochs

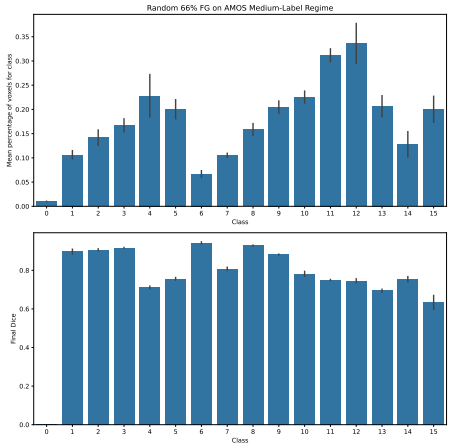
(f) High-Label - 500 Epochs

Figure D.8: Visualization of the difference of the percentage of voxels for all classes alongside Final Dice performance on AMOS in the Main setting from ClaSP PE to Random 66% FG trained with 200 & 500 epochs. Error bars denote the Standard Deviation.

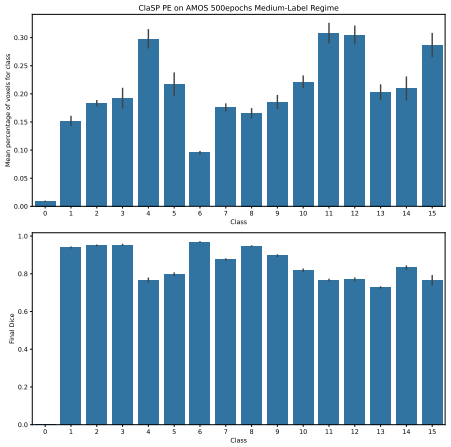


(a) Low-Label - 200 Epochs

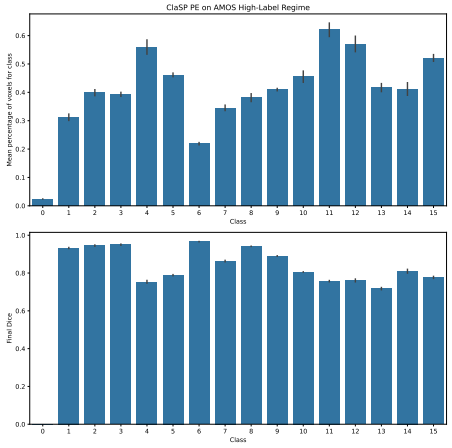
(b) Low-Label - 500 Epochs



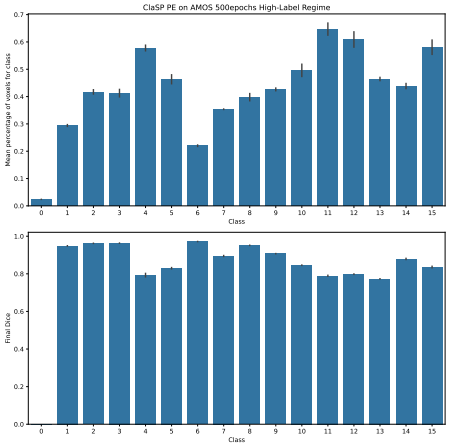
(c) Medium-Label - 200 Epochs



(d) Medium-Label - 500 Epochs



(e) High-Label - 200 Epochs



(f) High-Label - 500 Epochs

Figure D.9: Visualization of the percentage of voxels for all classes alongside Final Dice performance on AMOS in the Main setting from ClaSP PE trained with 200 & 500 epochs. Error bars denote the Standard Deviation.

D.5 Guidelines for Real-World Deployment of ClaSP PE

In the following, we provide details on the systematic selection of query patch size and query budget parameters for applying ClaSP PE to unseen datasets. The parameters that we obtain for the roll-out datasets are provided in table D.1. Despite our extensive validation, demonstrating the generalization capabilities, we can not guarantee good model performance beyond the tested settings, especially for lower fractions of annotated data.

Query Size Selection We recommend to normalize the query size based on the number of foreground classes in the dataset to optimally leverage the performance gains through class-stratified sampling. In our roll-out experiments, we calculate the total query budget (per AL cycle) by counting contribution of 50 or 100 patches per class, depending on the task complexity of segmenting certain target structures. Specifically, we use a query budget contribution of 100 for classes where we expect higher variance, such as the tumor class compared to the liver class in LiTS.

Number of AL Loops In our experiments we performed five AL loops. Since AL performance typically improves or remains stable relative to random strategies in later stages or with larger annotation budgets, extending the number of AL loops is generally safe and may further improve performance. We hypothesize that leaving the value for β (inverse power noising strength) after the 5th loop at 100 should be sufficient for ClaSP PE, as the segmentation model should by then be able to effectively exploit its understanding of the task. Crucially, we do not advise to reduce the number of AL loops.

Starting Budget Selection The starting budgets should be selected with a mix of completely random patches and a random class balanced selection of patches. We used a factor of 33% of patches which surely feature foreground. In practice, the patches which surely feature foreground can be simply obtained by going into random images and just selecting some patches featuring target structures of interest while counting the amount of patches to ensure that they are somewhat class balanced.

Query Patch Size Selection For the size of the query patches, we recommend choosing the median size of the target structures, in order to obtain representative samples. This is realizable in practice, as an estimation of the object sizes can typically be done efficiently, without necessitating the availablity of fine-grained annotations. In the benchmarking scenario, we proceed as follows: First, we compute the median bounding box size per class based on the largest connected component per image. Then, we take the overall median bounding box size across classes. An exception is the LiTS dataset, where we only consider the tumor class (omitting the liver class), as the liver is significantly larger while the tumor structures are of particular interest.

D.6 Roll-Out Results

Detailed results for the roll-out study in section 8.3.2 are provided in table D.6. The corresponding PPM is shown in fig. 8.6.

Table D.6: Fine-grained Results for the Roll-Out Scenario. Higher values are better, and colorization goes from dark green (best) to white (worst) with linear interpolation. AUBC and Final Dice are reported with a factor (×100) for improved readability. AUBC, Final, and FG-Eff can only be directly compared for each Label Regime on each dataset. The respective dataset characteristics are detailed in table D.1.

Dataset	LiTS			WORD			Tooth Fairy 2			MAMA MIA		
Label Regime	Roll-Out			Roll-Out			Roll-Out			Roll-Out		
Metric	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff	AUBC	Final Dice	FG-Eff
Query Method												
Random	51.23 ± 1.21	52.38 ± 2.21	46.25 ± 45.84	77.35 ± 1.04	78.03 ± 0.88	3.66 ± 0.25	61.83 ± 0.25	64.32 ± 0.38	11.88 ± 0.18	55.23 ± 2.06	58.24 ± 1.90	39.13 ± 209.13
Random 66% FG	48.63 ± 1.22	50.05 ± 1.32	1.27 ± 0.15	78.19 ± 0.34	78.25 ± 0.15	1.34 ± 0.02	65.30 ± 0.28	68.61 ± 0.15	10.85 ± 0.17	44.38 ± 3.68	45.10 ± 5.64	-4.67 ± 0.91
Predictive Entropy	57.81 ± 1.17	65.38 ± 2.76	38.94 ± 4.50	78.43 ± 0.07	78.96 ± 0.28	0.91 ± 0.00	66.65 ± 0.60	71.97 ± 0.08	16.25 ± 0.60	59.07 ± 4.15	64.74 ± 2.42	9.43 ± 2.08
ClaSP PE	60.30 ± 1.74	65.80 ± 1.47	39.60 ± 12.95	78.27 ± 0.41	78.42 ± 0.17	1.33 ± 0.02	67.32 ± 0.37	71.49 ± 0.17	20.07 ± 0.43	63.85 ± 1.58	68.62 ± 1.36	57.36 ± 407.92

D.7 Limitations

Benchmark overfitting. ClaSP PE was developed on the nnActive benchmark and therefore carries the risk of over-optimization. However, the general strong performance on the Roll-Out Study against the Predictive Entropy and Random FG 66% FG, which were the other best performing methods on the nnActive benchmark, shows its generalization capabilities to novel scenarios. We also wish to highlight that virtually all AL methods face the danger of being overdesigned for a specific benchmark as they necessitate design decisions that need to be evaluated empirically (J. Shi et al., 2024; Föllmer et al., 2024; Gaillochet et al., 2023b; Vepa et al., 2024). The combined evaluation on the nnActive Benchmark and Roll-Out study, which to our knowledge is the most comprehensive to date, mitigates the risk of benchmark-specific overfitting relative to earlier approaches. Further, the performance of ClaSP PE on the nnActive benchmark suggests generalization capabilities beyond our conservative Guidelines for Real-World Deployment. We encourage future benchmarking efforts of AL methods for 3D biomedical segmentation, demonstrating their generalization capabilities on novel datasets separate from method development, thereby reducing potential conflicts of interest.

Dependency on model predictive capacity. ClaSP PE relies on the model producing sufficiently accurate multi-class segmentations, since these predictions underpin the stratified query selection. In the low-Label Regime of the AMOS dataset (see section 8.3.1 Query Design), we observed that limited initial labels can result in inadequate segmentation quality, reducing the effectiveness of stratification. Our final recommended guidelines for using ClaSP PE mitigate this risk by using a query size based on the number of classes that is most likely to lead to a sufficient initial segmentation quality, as exemplified by the results in the Roll-Out study. Moreover, the use of pre-trained foundation models may further improve early-stage segmentation quality (Gupte et al., 2024).

Economic trade-offs of AL. We wish to emphasize that AL inherently represents a wager with the aim of reducing the overall cost of building an ML pipeline where compute cost is better against an expected reduction in annotation effort (Settles, 2011). The decision whether to employ AL or not is an economic question dependent on multiple factors, such as annotation cost and computational resources. In this work, we demonstrate that ClaSP PE within the nnActive Framework shows strong evidence to reduce the annotation cost when employing AL in a wide variety of settings. However, the cost of employing AL needs to be evaluated in comparison to the expected gains, which is outside the scope of this work and represents a fruitful direction for future research.

Comparison to more complex AL baselines. While more sophisticated AL strategies (s.a. Hübotter et al. (2024) and Föllmer et al. (2024)) could in principle yield similar gains, there is currently little evidence that they can be made practical in our setting. In particular, extending such methods to 3D biomedical image segmentation remains an unsolved challenge: querying 3D patches instead of 2D slices while integrating the full complexity of state-of-the-art segmentation pipelines poses substantial computational and algorithmic hurdles. Our focus here was therefore on designing an approach that is robust and easily deployable across new datasets, leaving the open problem of adapting more advanced AL techniques to the 3D domain for future work.

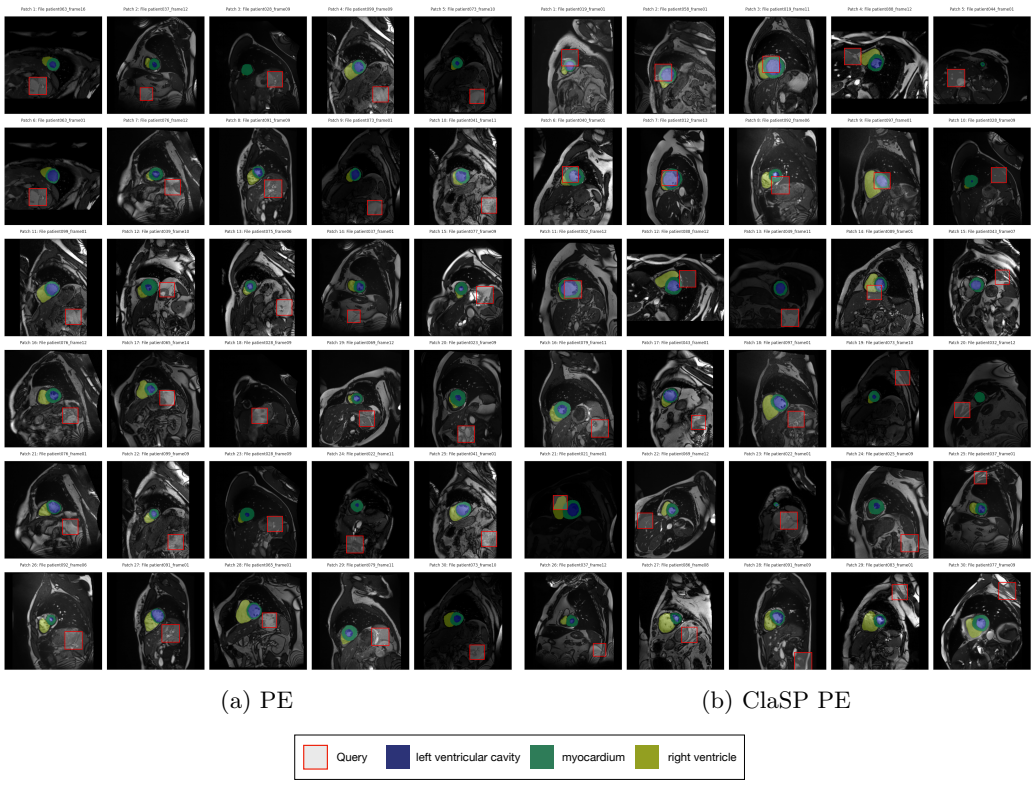
Hyperparameters of ClaSP PE. We addressed the need for a class-balanced dataset and hard-to-predict cases, and early-stage diversification and later exploitation through careful balancing of the stratified sampling ratio α and a pre-defined scheduling for power-noising of β. We empirically rigorously validate these modifications to ensure overall benefits on a wide variety of datasets based on a large-scale benchmark for development and an evaluation on held-out roll-out datasets. However, our results also show that this exact setup is not always the optimal solution, and it might be even more favorable to have heuristics to adapt both α and the scheduling β of beta based on dataset characteristics and other confounding information, such as model performance.

D.8 Qualitative Results

In this section, we provide additional qualitative results to demonstrate the effects of the class stratification used in ClaSP PE, as compared to standard Predictive Entropy.

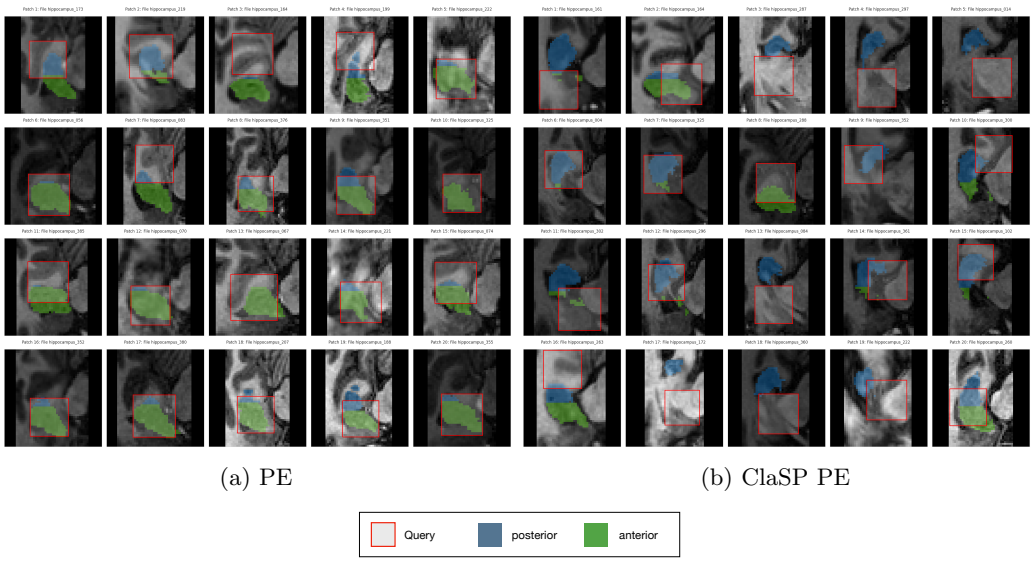
D.8.1 Query Visualization

In figs. D.10 to D.13, we provide exemplary visualizations of the queried patches of PE and ClaSP PE after the first AL loop on all nnActive benchmark datasets on the low-Label Regime using the main settings. For ACDC (fig. D.10), the stratification of ClaSP PE leads to a more diverse query selection and more foreground being queried. Further, ClaSP PE mitigates the risk of an overly focus on prominent classes, such as the posterior hippocampus (fig. D.11), the tumor class on KiTS (fig. D.12), or the liver class on AMOS (fig. D.13).



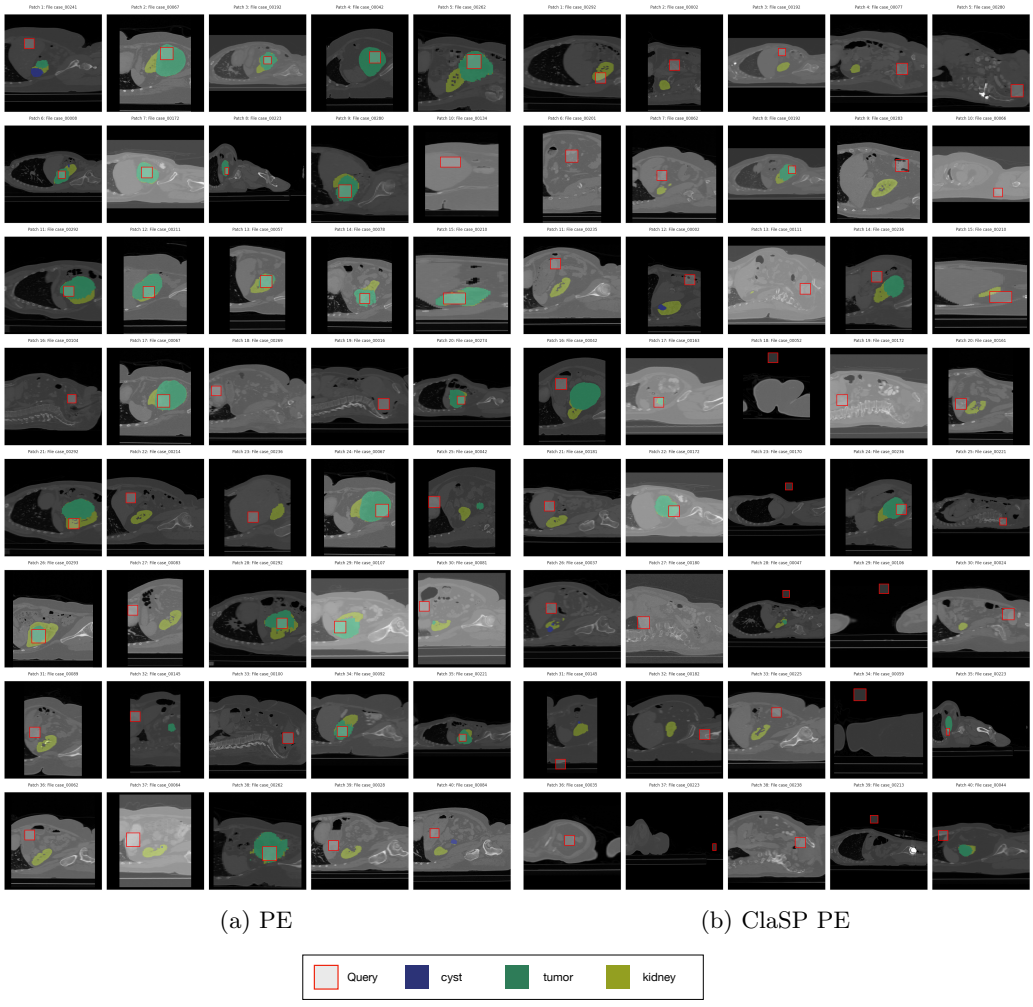
(a) PE (b) ClaSP PE

Figure D.10: Exemplary visualization of the queried patches using PE (a) and ClaSP PE (b) after the first AL loop on the ACDC dataset (same seed to ensure comparability). For 2D visualization, we selected the center slice of the 3D patches. Best viewed zoomed in.



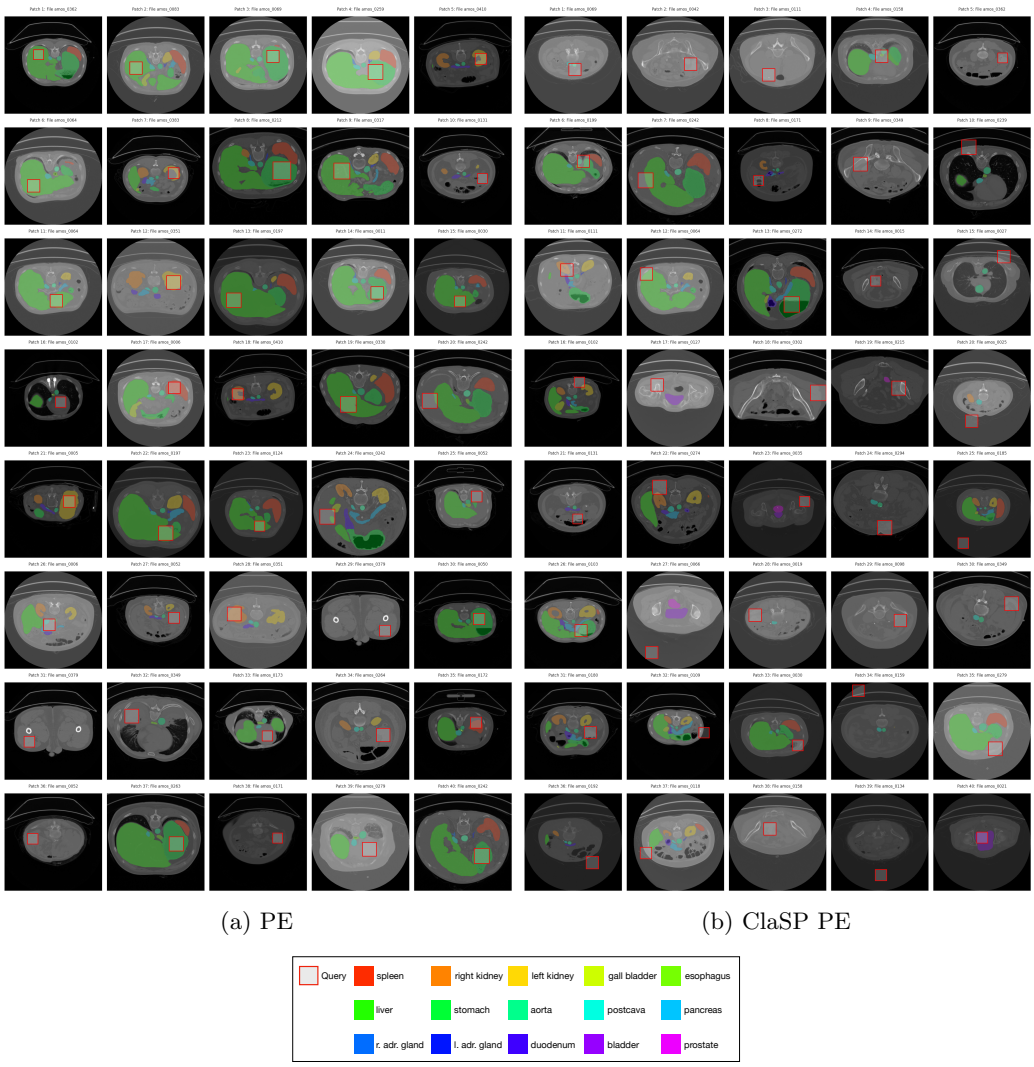
(a) PE (b) ClaSP PE

Figure D.11: Exemplary visualization of the queried patches using PE (a) and ClaSP PE (b) after the first AL loop on the Hippocampus dataset (same seed to ensure comparability). For 2D visualization, we selected the center slice of the 3D patches. Best viewed zoomed in.



(a) PE (b) ClaSP PE

Figure D.12: Exemplary visualization of the queried patches using PE (a) and ClaSP PE (b) after the first AL loop on the KiTS dataset (same seed to ensure comparability). For 2D visualization, we selected the center slice of the 3D patches. Best viewed zoomed in.



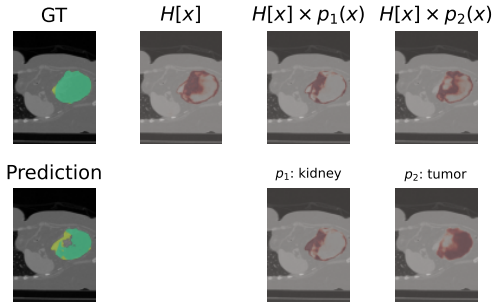
(a) PE (b) ClaSP PE

Figure D.13: Exemplary visualization of the queried patches using PE (a) and ClaSP PE (b) after the first AL loop on the AMOS dataset (same seed to ensure comparability). For 2D visualization, we selected the center slice of the 3D patches. Best viewed zoomed in.

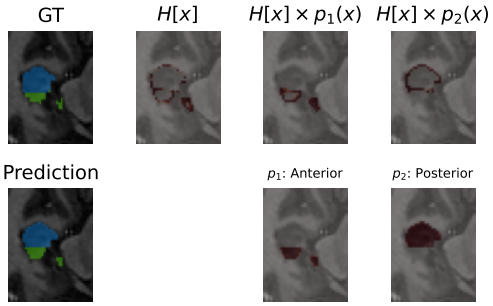
D.8.2 Stratification Visualization

Figure D.14 illustrates the class-wise stratification defined in eq. (D.1), based on predictive entropy. For this example, we use models from the first loop of the Low-Label Regime in the main nnActive benchmark setting.

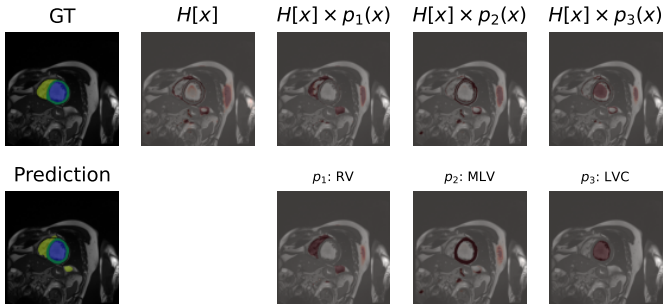
The figure shows that this stratification shifts the regions of high uncertainty for each class toward the areas where the model's predictions indicate these classes are present.



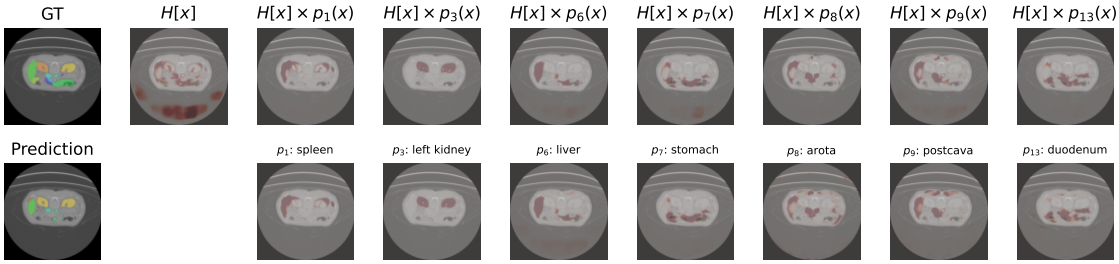
(a) KiTS (case 67).



(b) Hippocampus (case 8).



(c) ACDC (patient 3, frame 1).



(d) AMOS (case 4).

Figure D.14: Visualizations of the stratification mechanism of ClaSP PE for exemplary cases of each nnActive benchmark dataset. For each predicted class in the displayed slice, the class probabilities $p_i(x)$ as well as the weighted entropy maps $H[x] \times p_i(x)$ are shown, which lay the basis of the stratified querying. The colormaps are rescaled for each individual image. To avoid outliers distorting the color mapping, we clip high values at the 98% quantile of the data.