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

vorgelegt von
Zharov, Yaroslav, Dipl.-Ing.
aus Moskau

Tag der mündlichen Prüfung:

# Relaxing Supervision Requirements for Tomographic Data Analysis with Machine Learning

Prof. Dr. Vincent Heuveline:

# Preface

The work on this thesis has led to several publications both in peer-reviewed venues and on preprint servers.

### Peer-reviewed publications

1. Yaroslav Zharov, Evelina Ametova, Rebecca Spiecker, Tilo Baumbach, Genoveva Burca, and Vincent Heuveline (July 2023). "Shot noise reduction in radiographic and tomographic multi-channel imaging with self-supervised deep learning". In: *Optics Express* 31.16, p. 26226. ISSN: 1094-4087. DOI: 10.1364/OE.492221

2. Rebecca Spiecker, Pauline Pfeiffer, Adyasha Biswal, Mykola Shcherbinin, Martin Spiecker, Holger Hessdorfer, Mathias Hurst, Yaroslav Zharov, Valerio Bellucci, Tomáš Faragó, Marcus Zuber, Annette Herz, Angelica Cecilia, Mateusz Czyzycki, Carlos Sato Baraldi Dias, Dmitri Novikov, Lars Krogmann, Elias Hamann, Thomas van de Kamp, and Tilo Baumbach (Dec. 2023b). "Dose-efficient in vivo X-ray phase contrast imaging at micrometer resolution by Bragg magnifiers". In: *Optica* 10.12, p. 1633. ISSN: 2334-2536. DOI: 10.1364/OPTICA.500978

3. Jwalin Bhatt, Yaroslav Zharov, Sungho Suh, Tilo Baumbach, Vincent Heuveline, and Paul Lukowicz (Apr. 2023). "A Knowledge Distillation Framework for Multi-Organ Segmentation of Medaka Fish in Tomographic Image". In: *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)*. IEEE, pp. 1–5. ISBN: 978-1-6654-7358-3. DOI: 10.1109/ISBI53787.2023.10230689

### Preprints

1. Yaroslav Zharov, Alexey Ershov, Tilo Baumbach, and Vincent Heuveline (Mar. 2022). "Using the Order of Tomographic Slices as a Prior for Neural Networks Pre-Training". In: URL: http://arxiv.org/abs/2203.09372

2. Yaroslav Zharov, Tilo Baumbach, and Vincent Heuveline (Mar. 2023). "Optimizing the Procedure of CT Segmentation Labeling". In: URL: http://arxiv.org/abs/2303.14089

# *Abstract*

**Relaxing Supervision Requirements for Tomographic Data Analysis with Machine Learning**

In this doctoral thesis, the power and potential of advanced imaging techniques, specifically Tomographic Imaging (hereinafter tomography), are explored in an era characterized by the rapid growth of data and the critical need for effective analysis strategies. This work engages with different modalities, such as but not limited to parallel beam X-ray Computed Tomography (CT), and Magnetic Resonance Imaging (MRI). The research is centered around the incorporation of machine learning models, deep learning in particular, to optimize the analysis of tomography scans across various domains, including biology, medicine, and material sciences. This is achieved by navigating the primary challenges associated with the utilization of tomography, namely image preprocessing, data labeling, and model training. This work is organized as a series of chapters, consequently covering those topics in the order in which the proposed techniques would be applied in a practical pipeline of the data analysis.

In Chapter 3 this work explores the applicability of the Noise2Noise denoising technique to the multi-channel imaging datasets, particularly those with significantly reduced Signal-to-Noise Ratio (SNR). Utilizing the self-supervised denoising approach for datasets for biological and material sciences, significant improvements in image quality have been achieved, or, equivalently, the possibility to reduce exposure time has been shown while maintaining image quality.

Chapter 4 of the thesis details the optimization of dataset preparation procedures for training neural networks, specifically concerning tomography segmentation tasks. The study conducted on several openly available medical datasets unravels the critical elements of a useful dataset: quality, diversity, and completeness. It further proposes an optimized labeling procedure that balances these virtues, aiming to deliver the best dataset with minimal effort.

Chapter 5 introduces a novel self-supervised pre-training technique for biomedical tomography called *SortingLoss*. Its underlying principle is the utilization of the inherent order of slices in a tomography scan volume to pre-train a neural network. This method has been evaluated on medical tomography of lungs affected by COVID-19 and high-resolution full-body tomography of model organisms (Medaka fish) showing lower computational complexity while maintaining results on par with more complex but general approaches.

Lastly, Chapter 6 presents a self-training framework for multi-label segmentation, therefore marking the last stage of the data analysis. The pseudo-labeling method, complemented by a novel Quality Classifier technique to select the best pseudo-labels, and pixel-wise knowledge distillation, has led to improved segmentation performance when tested on the dataset of the Medaka fish brain segmentation.

In sum, this thesis explores the profound potential of integrating advanced computer vision and machine learning tools in the application of tomography imaging. It proposes novel solutions to existing challenges and applies existing techniques in novel circumstances, aiming to remove the burden of manual analysis from the area experts.

In dieser Doktorarbeit werden die Kraft und das Potenzial fortschrittlicher bildgebender Verfahren, insbesondere der Tomographischen Bildgebung (im Folgenden Tomographie), in einer Ära untersucht, die durch das schnelle Wachstum von Daten und die dringende Notwendigkeit effektiver Analysestrategien gekennzeichnet ist. Diese Arbeit beschäftigt sich mit verschiedenen Modalitäten, wie aber nicht beschränkt auf Parallelstrahl-Röntgen-Computertomographie (CT) und Magnetresonanztomographie (MRT). Die Forschung konzentriert sich auf die Einbeziehung von maschinellen Lernmodellen, insbesondere des tiefen Lernens, um die Analyse von Tomographie-Scans in verschiedenen Bereichen, einschließlich Biologie, Medizin und Materialwissenschaften, zu optimieren. Dies wird erreicht, indem die primären Herausforderungen im Zusammenhang mit der Nutzung der Tomographie, nämlich Bildvorverarbeitung, Datenbeschriftung und Modellschulung, bewältigt werden. Diese Arbeit ist als eine Reihe von Kapiteln organisiert und deckt folglich diese Themen in der Reihenfolge ab, in der die vorgeschlagenen Techniken in einer praktischen Datenanalyse-Pipeline angewendet würden.

In Chapter 3 untersucht diese Arbeit die Anwendbarkeit der Noise2Noise Entstörungstechnik auf Multikanal-Bildgebungs, insbesondere solche mit deutlich reduziertem Signal-Rausch-Verhältnis (SNR). Durch die Verwendung des selbstüberwachten Entstörungsansatzes für Datensätze aus den Bereichen Biologie und Materialwissenschaften wurden signifikante Verbesserungen der Bildqualität erzielt oder, anders ausgedrückt, die Möglichkeit gezeigt, die Belichtungszeit zu reduzieren und dabei die Bildqualität beizubehalten.

Chapter 4 der Arbeit beschreibt die Optimierung von Verfahren zur Datensatzvorbereitung für das Training neuronaler Netzwerke, insbesondere im Hinblick auf CT-Segmentierungsaufgaben. Die Studie, durchgeführt an mehreren öffentlich verfügbaren medizinischen Datensätzen, entfaltet die kritischen Elemente eines nützlichen Datensatzes: Qualität, Vielfalt und Vollständigkeit. Sie schlägt weiterhin ein optimiertes Beschriftungsverfahren vor, das diese Tugenden ausbalanciert und darauf abzielt, den besten Datensatz mit minimalem Aufwand zu liefern.

Chapter 5 führt eine neuartige selbstüberwachte Vor-Trainingstechnik für biomedizinische CTs ein, genannt *SortingLoss*. Ihr zugrundeliegendes Prinzip ist die Nutzung der inhärenten Reihenfolge von Schnitten in einem CT-Scanvolumen, um ein neuronales Netzwerk vorzutrainieren. Diese Methode wurde an medizinischen CTs von Lungen, die von COVID-19 betroffen sind, und an hochauflösenden Ganzkörper CT von Modellorganismen (Medakafische) getestet und zeigt eine geringere Rechenkomplexität, während sie Ergebnisse auf Augenhöhe mit komplexeren, aber allgemeineren Ansätzen beibehält.

Schließlich stellt Chapter 6 einen Selbsttrainingsrahmen für die Mehrfachetikettensegmentierung vor und markiert somit die letzte Stufe der Datenanalyse. Die Pseudobeschriftungsmethode, ergänzt durch eine neuartige Quality Classifier Technik zur Auswahl der besten Pseudobeschriftungen, und pixelweise Wissensdestillation haben zu verbesserten Segmentierungsleistungen geführt, als sie am Datensatz der Medakafisch-Gehirnsegmentierung getestet wurden.

Zusammenfassend erforscht diese Dissertation das tiefe Potenzial der Integration fortschrittlicher Computer Vision und maschineller Lerntechnologien in der Anwendung der CT-Bildgebung. Sie schlägt neuartige Lösungen für bestehende Herausforderungen vor und wendet bestehende Techniken in neuen Umständen an, mit dem Ziel, die Last der manuellen Analyse von den Fachexperten zu nehmen.

# *Acknowledgements*

A profound thank you to my father, whose love for the elegance of mathematics and engineering kindled a similar fire within me. My wife deserves immense gratitude. Through life's ups and downs, she remained my bedrock of support, weathering the storm with remarkable strength and grace. I also thank Professors Baumbach and Heuveline, who nurtured my curiosity, allowing me the freedom to explore the topic at my own pace.

Each one of you played a vital role in this journey. Your support has been invaluable, and for that, I am wholeheartedly grateful. Thank you.

# Contents

# List of Abbreviations and Terminology

**Radiography (X-ray Imaging)** A non-invasive diagnostic technique employing X-rays to generate images representing the internal composition of an object or body, facilitating the examination of structural integrity or pathology without requiring direct access.

**CT (Computed Tomography)** An advanced imaging modality that synthesizes multiple X-ray projections taken from diverse angles to create volumetric cross-sectional images, enabling detailed internal visualization without physical sectioning.

**MRI (Magnetic Resonance Imaging)** A radiological technique that utilizes strong magnetic fields and radiofrequency pulses to produce detailed images of internal body structures, highlighting physiological processes and anatomical details.

**Biomedical Imaging** The interdisciplinary field that combines elements of engineering, biology, and medicine to visualize internal structures for diagnostic, therapeutic, and research purposes, encompassing a broad spectrum of imaging modalities.

**ML (Machine Learning)** A branch of artificial intelligence focused on developing algorithms that enable computers to learn from and make predictions or decisions based on data, without being explicitly programmed for specific tasks.

**DL (Deep Learning)** A subset of machine learning characterized by the use of deep neural networks with multiple layers of processing units, facilitating the modeling of complex patterns and high-level abstractions in data.

**NN (Neural Network)** A computational architecture inspired by the neural networks in the human brain, consisting of interconnected units (neurons) that process information using a layered approach, adaptable to a wide range of tasks through learning.

**Image Segmentation** The process of dividing a digital image into multiple segments (regions or pixels) to simplify or change its representation into a more meaningful format, facilitating easier analysis or interpretation.

**k-NN (k-Nearest Neighbors)** A non-parametric method in machine learning for classification and regression that estimates the likelihood of a data point belonging to one class or another based on the closest data points in the feature space.

**FBP (Filtered Back Projection)** A reconstruction algorithm widely used in CT imaging to produce two-dimensional images from the raw data acquired from multiple angles around the object, applying filters to correct distortions and enhance quality.

**Sinogram** A data representation derived from the raw measurements collected during a computed tomography (CT) scan, depicting the intensity of transmitted radiation as a function of the projection angle and the position of the radiation source. Sinograms serve as the foundation for algorithms such as filtered back projection (FBP) to reconstruct cross-sectional images from the radiographic projections.

**SNR (Signal-to-Noise Ratio)** A quantitative measure comparing the level of a desired signal to the background noise, crucial in determining the quality and clarity of an image or signal, with higher values indicating less noise and clearer detail.

**PR (Phase Retrieval)** An algorithmic approach in imaging science for extracting phase information from intensity patterns, indispensable in fields where direct phase measurement is challenging, enhancing image contrast and resolution in non-invasive imaging.

**ToF (Time-of-Flight)** A technique for estimating distance based on the travel time of a signal (light, sound, or particles) from a source to an object and back to a detector, applied in 3D imaging, LIDAR, and other spatial measurement contexts.

# Chapter 1

# Introduction

Analyzing specific instances, whether they are biological specimens like wasps and mice, physical entities such as crystals and liquids, or historical artifacts like ancient books and pottery, is a fundamental step in advancing our understanding of the world across natural and applied sciences. Earlier, e.g., in biology, scientists explored the structures of an organism with the naked eye, then with a microscope. However, not only light microscope have its limitations on the possible resolution (Evennett and Hammond, 2004), but it also has two intrinsic limitations: the ability to image either the exterior of an opaque sample or the projection of a translucent sample, never the full 3D information. The quasi-3D information became obtainable with light sheet microscopy, where the sample is illuminated with a thin and wide laser beam (sheet), and the microscope captures the luminescence of the illuminated sample layer (Huisken et al., 2004). The ability to image samples opaque to visible light came with the discovery of X-rays, which allowed the exploration of the intrinsic structure of such samples without physical intervention. The next step in the process of the sample's inner structure discovery was the Computed Tomography (CT) technique, which is able to produce a dense 3D image of the sample, a detailed, solid-filled volumetric internal representation, without any interventions other than exposure to X-ray radiation. While exposure to the radiation itself is yet a problem to be solved, especially for living creatures, this technique allowed for a new level of understanding of the microworld (Keklikoglou et al., 2021).

The CT is applied with various modifications in a multitude of knowledge domains. It has become an essential technique for medicine. E.g., as Power et al., 2016 claims, it has drastically reduced the need for exploratory surgery. It also increased the quality of early disease detection and became a popular screening technique (Bin Saeedan et al., 2016; MacMahon et al., 2005). When the COVID-19 epidemic stroke, CT became one of the important steps to precisely diagnose and understand the disease development stages (Kwee and Kwee, 2020, Figure 1.1).

In biology, applications of CT vary widely. It allowed a deep dive into otherwise inaccessible domains such as insect anatomy (Van De Kamp et al., 2011) or preserved fossils (Kamp et al., 2018, Figure 1.2). The nano-scale CT allowed Bradley, Robinson, and Yusuf, 2017 to image the cell growth process on polymer scaffolds with unprecedented resolution. The *in-vivo* CT allowed Moosmann et al., 2013 to discover cell behavior during the gastrulation phase of the *Xenopus laevis* embryo development. The high level of process automation allowed Weinhardt et al., 2018 to image hundreds of samples and create a dataset large enough for the statistical analysis of the morphology features of the *Oryzias latipes* fish (also known as Medaka fish).

In material sciences, CT is brought to the production, *e.g.*, quality assurance. It allowed Camattari et al., 2020 to conduct advanced crystallography, and (Vásárhelyi et al., 2020) to investigate dense materials in a non-destructive way. Being coupled with computer vision, CT allowed Fuchs, Kröger, and Garbe, 2021 to develop

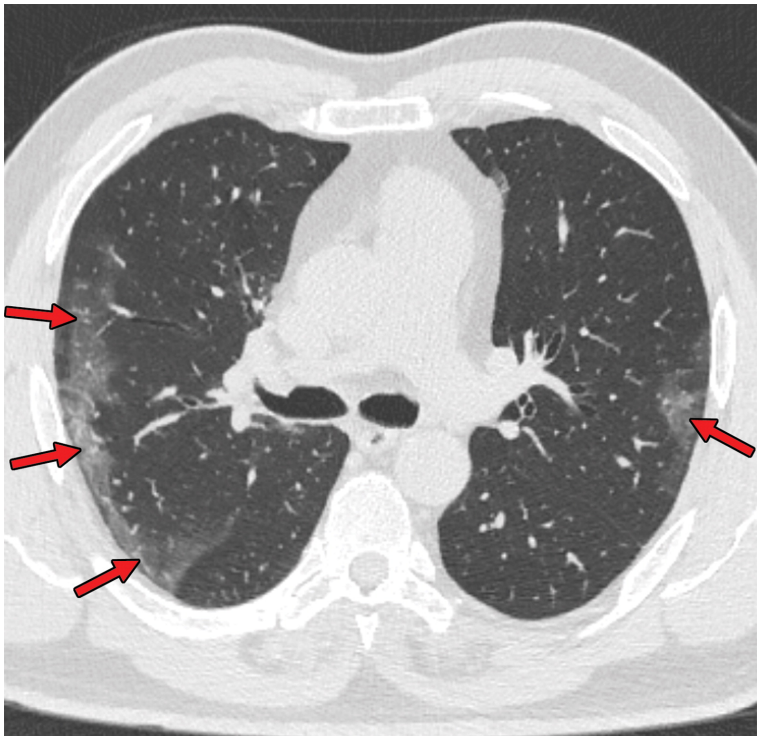


FIGURE 1.1: Demonstration of typical COVID-19 development features in lungs CT, as presented by (Kwee and Kwee, 2020)

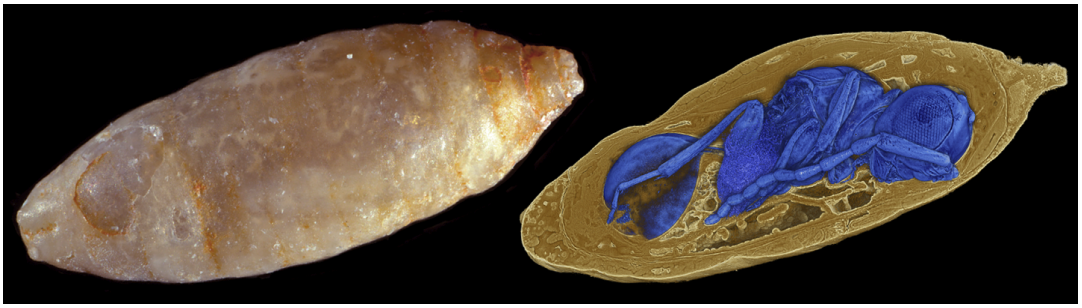


FIGURE 1.2: Photo (left) and 3D model built based on the CT volume (right) of a mineralized fossil, discovering the parasitoid wasp morphology, as presented by (Kamp et al., 2018)

industry-level techniques for defect detection.

Even though Computed Tomography fueled advances in multiple research disciplines, it has its own problems. For example, as Moosmann et al., 2013 states the ability of the penetrative X-Rays to destruct the living samples limits the possible time of imaging. As another example, as Weinhardt et al., 2018 and Fuchs, Kröger, and Garbe, 2021 show, CT requires advanced analysis techniques to thrive in the age of big data analysis.

Machine learning models have become an essential tool in CT data analysis due to the vast amounts of data produced by CT scanners. These models can be trained to analyze CT images with accuracy and speed inaccessible to humans. By automating the analysis process, machine learning can greatly improve both the efficiency and accuracy of CT data analysis, allowing e.g., medical experts to make more informed decisions about patient care, without tediously measuring needed parameters on the CT; biology experts, to acquire statistical data on their subject organisms without manually segmenting tens and hundreds of volumes; material experts, to discover patterns and abnormalities even in noisy data, which is painstakingly hard to read by a bare eye.

One specific example of ML for CT data analysis is the use of deep learning algorithms for the automated segmentation of organs and tissues. In biomedical CT scans, organs and tissues can appear as closely resembling each other, making it difficult to distinguish between them with simple tools. Deep learning algorithms can be trained on large datasets of annotated CT scans to learn how to precisely segment and label different organs and tissues. This automated segmentation can then be used for further analysis, from treatment planning for patients to quality assurance for microprocessors. For example, in medicine, Avetisian et al., 2020 demonstrated the ability to surpass the medical expert quality in the classification of stroke type via segmentation. This result, applied at scale, can help earlier, and more precise treatment for diseases, leading to increased life expectancy.

To summarize, CT became a key technique in many areas, from medicine and biology to material research. However, it still has its bottlenecks slowing the transfer from imaging to conclusions and discoveries. Machine Learning techniques and Deep Learning, in particular, became established techniques to widen those bottlenecks, or sometimes even surpass them completely. However, to be applicable they require, large sets of precisely prepared data and sometimes domain-specific knowledge utilized during the training. The aim of this work is to contribute towards overcoming these bottlenecks with the tools provided by advanced computer vision and machine learning. Further in this chapter, I will briefly introduce the basics of Computed Tomography and Machine Learning.

## 1.1 Computed Tomography

Despite this work being mainly dedicated to machine learning, it is important to understand the peculiarities of the underlying imaging techniques. In this section, I will first describe the practical process of CT image formation. Then, I will discuss the important differences between different imaging domains, techniques, and samples.

### 1.1.1 Computed Tomography Acquisition

To obtain a CT image, we need three physical components:

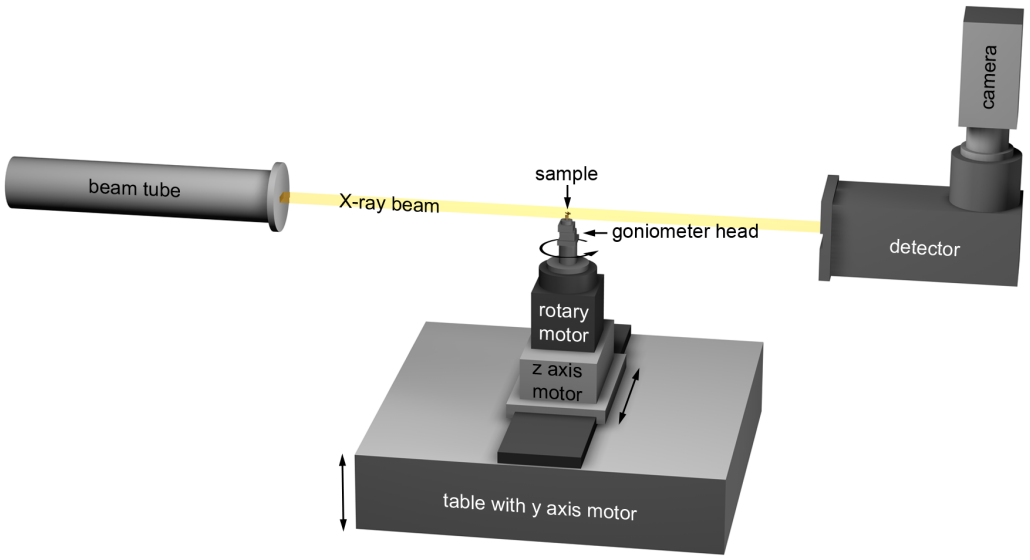



FIGURE 1.3: Imaging station for the Synchrotron-based Computed Tomography, as presented by Kamp, 2011. The beam tube represents the source here, the sample is positioned on a rotational stage in the middle, and the detector is on the right.

**Source** of the penetrating rays of some sort. For example, an X-Ray tube, or a synchrotron light source.

**Detector** – a device capable of measuring the spatial distribution of the X-rays' intensity. Usually, some sort of digital camera, either with the matrix sensitive to the required energy spectrum (so-called direct detectors) or a classical camera pre-pended with a scintillator converting the X-rays to the visible spectrum (so-called indirect detectors).

**Sample** – the object of imaging. The sample should be translucent to the X-rays, yet not completely transparent. The exact limitations are dependent on the source and camera properties (the penetrative ability of the rays, the resolution of the camera, etc).

I show a schematic setup in the Figure 1.3. The sample is situated between the source and the detector in such a way, that the beam emitted from the source interacts with the sample and is then captured by the detector. By interaction between the beam and the sample we typically understand the attenuation of the beam. Each pixel of the detector, in this case, captures the energy of the arrived beam. Either sample or source-detector system should be able to rotate with high precision, to capture the *projections* of the sample from different angles. The axis, around which the system rotates is called *tomographic rotation axis*. Typically, during a *scan*, a set of projections is taken evenly spaced across 180° of rotation. We will call the rotation angle of each projection with respect to the starting position a *projection angle*. The amount of projections taken during the scan is a parameter of the imaging process and is set w.r.t. the properties of the system and the sample. The high amount of projections requires precise positioning of the sample and increases the exposure time of the sample, but provides a clearer, less noisy image.

The acquired projections should then be treated to avoid excessive noise and amplify the signal. There are at least two corrections that are typically applied, the flat and dark field corrections.

**The flat field correction** is used to remove the noise introduced by the source. The source rarely covers the whole field of view evenly, and the beam is frequently disturbed by the imperfections of the components. The flat field itself is, therefore, captured with the source turned on, but without the sample in the field of view.

**The dark field correction** is used to compensate for the artifacts of the camera itself, like the peculiarities of the matrix. The dark field is captured with the source turned off.

If we have the projection $I$, flat field $F$, and dark field $D$, the correction is by the Equation (1.1).

$$\hat{I} = \frac{(I - D)\bar{I}}{F - D} \tag{1.1}$$

, where $\bar{I}$ stands for average pixel value, and $\hat{I}$ for the corrected image. Hereinafter, I will always assume the images are always flat and dark field corrected.

These corrections are enough when we speak about the typical imaging regime, also called *absorption contrast*. In this regime, each projection pixel represents how much of the X-ray got absorbed going through the tissues. However, for thin objects, or for soft tissues, that do not absorb much, this regime doesn't provide much information. That's where the so-called *phase contrast* regime is used. This regime is based on the idea that despite the absorption of some materials being verlow, the speed of light is different for them. Therefore, the coherent wavefront is deformed while passing through the sample. Due to the coherence of this wavefront, in the locations where it is deformed, it forms the interference fringes that grow in size as the wavefront propagates further from through space (see Figure 1.4). The amplitude information on those fringes is captured by the detector after some propagation distance. However, instead of the typical absorption contrast projection, here we get a projection where only the borders between the materials are pronounced. To retrieve the image close to the absorption contrast, another correction is needed, on top of the dark and flat field corrections, the so-called *phase retrieval*. One of the popular ways to solve this task is the Paganin algorithm (Paganin et al., 2004), which is, simply speaking, a specially formulated low-pass filter, which works for small propagation distances, where the fringes have low overlap. After this correction, the resulting image resembles the one of the absorption contrast, but the materials with the low absorption become more pronounced.

After all corrections are done to the projections, the final step is to reconstruct the 3D volume. This step is based on the ability to reconstruct a 3D volume from a series of its 2D projections. This ability is backed by the work of Radon, 1986, where he demonstrated the reversibility of a specific integral transform matching a 2D function and set of its 1D projections on differently oriented axes (hereinafter the Radon transform). Given a 2D function $f(x_1, x_2)$, we introduce the transformed version of this function as $\hat{f}(\rho, \theta)$ – the function of a projection angle $\rho$ and position along the projection axis $\theta$ (see the Figure 1.5 for intuition). To find a value of the $\hat{f}$ at some point $(\rho^*, \theta^*)$, we need to take the integral of the function $f$ along the line $x_2 = tg(\theta^*)x_1 + \frac{\rho^*}{cos(\theta^*)}$.

The projections are stacked together, along the new axis, to constitute a 3D image (Figure 1.6). The axes of this 3D image are the concatenation axis $\theta$, and $x$ and $y$, the pair of axes corresponding to the two axes of each projection image. Speaking of

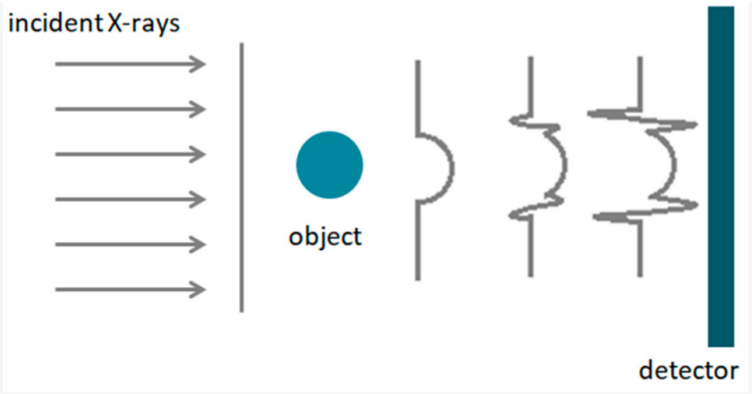



FIGURE 1.4: Sketch of propagation-based imaging setup as presented by Tao et al., 2021.

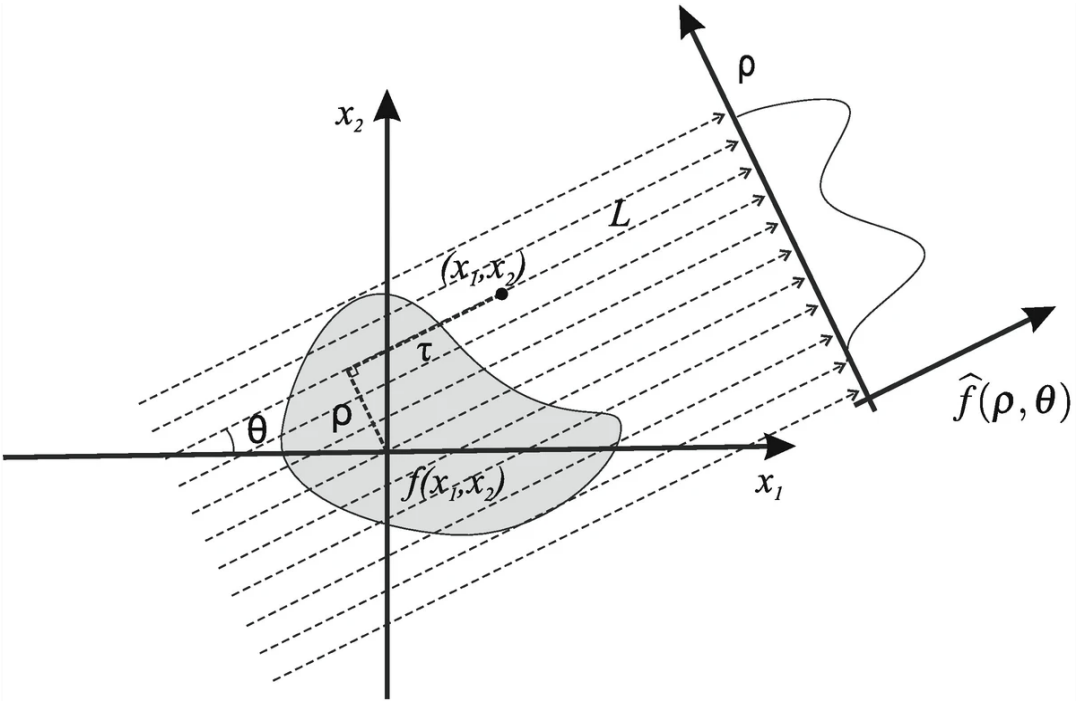



FIGURE 1.5: Visual explanation of the Radon transform performed from the 2D function $f(x_1, x_2)$, defined on the cartesian axes $x1, x2$ to another 2D function $\hat{f}(\rho, \theta)$, defined on the polar axes $\theta, \rho$. The second function can be viewed as an infinite collection of 1D functions, where each 1D function is a line integral of the inial function along the newly selected axis, rotated at the angle $\theta$ to the original $x_1$ axis, and the $\theta$ denotes the shift along the axis orthogonal to this newly selected one. The idea is presented in further details in the work by Protonotarios et al., 2021, where I took the plot from.

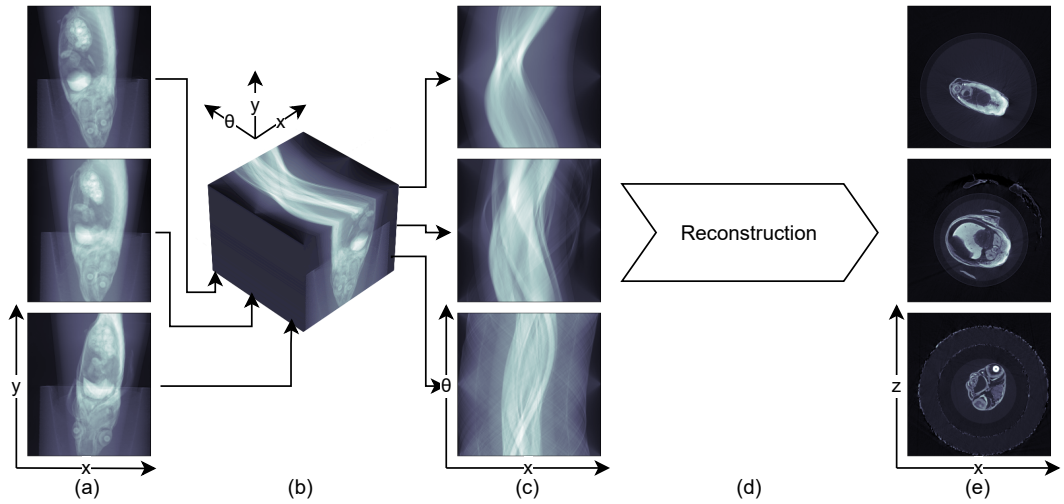



FIGURE 1.6: Obtaining the tomographic volume from a set of projections. A set of projections (a) is obtained for different projection angles. They are stacked into an array (b), along the new axis, representing the rotation angle. This array then is sliced along the plane orthogonal to the tomographic rotation axis, to obtain a set of sinograms (c). The reconstruction (d) (e.g., Filtered Back Projection) is then used, to obtain a set of slices (e) that make up the final volume.

connection with the Figure 1.5, the axis $\theta$ represents the same axis, the axis $x$, represents the axis $\rho$, and the axis $y$, represents the axis enumerating the tomographical slices, going along the tomographic rotation axis. This axis is orthogonal both to $x_1$ and $x_2$ on the Figure 1.5. We then slice this 3D image along the plane orthogonal to the $y$ axis to get a so-called *sinograms*. Due to the fact, that each sinogram in this stack is orthogonal to the tomographic rotation axis, all points of the object imaged on it, never leave one sinogram to appear on another. During rotations, points of the object form sinus-shaped lines, this is where the name comes from.

This sinogram is a sampled representation (since we have a finite number of projections) of a 2D slice of the sample but in the Radon space as $\hat{f}$. Using the reversibility of the Radon transform, we can reconstruct a slice of the final 3D volume, with axes $x$ and $z$ of sizes of both equal to the size of the horizontal size of the original projection. There are multiple algorithms to obtain this reconstruction. The simplest and the most popular one is the Filtered Back Projection (FBP) algorithm (Schofield et al., 2020).

However, the problem of inverting the Radon transform from the sampled noisy observations is ill-posed, which can lead to incorrect solutions in case of high noise or a low number of projections acquired. For extremely noisy or undersampled cases a variation of the reconstruction procedure called *iterative reconstruction* can be used instead of FBP. In this case, the reconstruction process is viewed as an optimization problem. The main benefit of the iterative reconstruction is the ability to impose additional restrictions on the solution. One typical example of such limitation is a restricted set of possible values in the reconstructed volume, which implies the limited amount of materials in the sample. Another option would be to enforce the smoothness of the resulting volume to mitigate excessive noise (Ametova et al., 2021b).

### 1.1.2 Imaging Modalities Overview

Radiography, CT, and biomedical imaging differ from photography, not only from the appearance point of view. In photography, despite some general spatial relations taking place, the general composition is not fixed, which, on the one hand, leads to a drastic difference between two photos displaying the same object, but on the other hand, helps models to delineate features of the objects from features of its surroundings. For, despite some works showing that models overly rely on texture (Geirhos et al., 2018) and can fail to recognize objects in unusual surroundings (Gupta et al., 2022), in general, the model trained to detect some objects on images can locate them (Selvaraju et al., 2016). In turn, for biomedical imaging, the controlled environment and similarity of the samples make the variability less prominent. For example, the heart of a living human being is always located near the lungs and is always present in chest radiography or CT. Therefore, it is impossible to train a model to classify radiography images of humans with regard to the existence of the heart, and the model should be able to locate it. Furthermore, fixed imaging conditions make space for spurious correlations, e.g., the work by Narla et al., 2018 shows how if the ruler appears on the skin photo, it is a strong sign of malignancy of the skin lesion imaged.

Other aspects are superposition, occlusion, and perspective. In photography, they occur altogether, requiring a model to adapt to drastically changing sizes of the objects or partial visibility. In radiography, while occlusion is generally possible (e.g., so-called bone shadows), it is more predictable and can be efficiently mitigated (Rajaraman et al., 2021). As the CT is a volumetric image, not only occlusion but also the superposition is excluded, apart from the image resolution issues. It means that the only possibility of some voxel being occupied by two different tissues or materials is for it to be spatially large enough. The perspective distortion can be mitigated in the CT image even when the cone beam projection is used. Furthermore, CT and radiography typically register the position of the imaging object, source, and detector. Therefore, the sample's size variability due to perspective not only can be mitigated mathematically but is of a smaller scale compared to photography.

The vast availability of photo cameras, the drastic difference between imaging conditions, and the popularity of photo-specific augmentations in Deep Learning allow training more generalizable models. Unlike photos, imaging protocol often strictly regulates radiography and CT imagery. Therefore, within one acquisition sequence on one device, the images of different samples may come out almost perfectly with the same image properties (contrast, brightness, noise, etc.), but the next sequence or a sequence imaged on a different device may come out dramatically different, lowering model performance. Combined with the lack of CT-specific augmentations, this leads to a search for more sophisticated transfer learning algorithms (Valverde et al., 2021).

Last but not least, it is important to note that not only do imaging modalities have a multitude of fundamental differences between them, but also the imaging type, setup, and analysis aims may vary the algorithm selection. For example, the medical MRI can be done on thousands of machines worldwide and has highly standardized protocols. But at the same time, there are only tens of synchrotron facilities worldwide that are capable of producing high-resolution images of living frog embryos, each being a one-of-a-kind device with setups and equipment different While large standardized samples (being biological or technical) can be physically pre-aligned, small biological samples (e.g., frog embryos) scanned with high resolution simply can not be perfectly aligned for the image to be taken and require post-processing.

While acquiring samples for morphology study requires visible structures, some material studies are more interested in the texture of the materials, leading to different imaging conditions. All this leads to the situation where a general dataset and pre-trained model (akin to the ImageNet by Russakovsky et al., 2015) is not only hard to achieve but also can have a questionable quality boost.

## 1.2 Machine Learning

With serial, high-throughput tomography, acquiring large datasets of CT 3D or even 4D images became possible. Which leads to the next bottleneck in the analysis problem–data processing. The current acquisition speed does not allow manual data analysis. The automation of the analysis has undergone significant progress, from routine automation to predictive models obtained by means of Machine Learning (ML).

Not only are those models able to free researchers from manual recurring labor, but they are also reproducible. Deep Learning (DL) techniques generate state-of-the-art results. Recent results demonstrate (Avetisian et al., 2020) that modern DL models can surpass expert quality. In this section, I will briefly review the ML approaches' tasks, models, and supervision regimes. A more thorough description can be found in works by Hastie, Tibshirani, and Friedman, 2009; Bishop, 2007; Murphy, 2012; Goodfellow, Bengio, and Courville, 2016.

### 1.2.1 Problems

Machine learning problems generally differ in terms of the inputs they expect and the outputs they produce in response. They could be roughly separated into three categories: classification, regression, and generation.

**Classification** stays for the cases, where for each input, the model should select one of the pre-defined possible classes. So, the model is a function that, given the data sample $x$ is expected to predict categorical value $y \in C$, where $C$ is a pre-defined set of classes, e.g., $C =$`{'cat', 'dog', 'human'}`. One typical example of classification is the prediction of the disease stage based on the patient's records.

**Regression** in turn, outputs a continuous value per input. So, given the sample $x$, the model should predict $y \in \mathbb{R}^n$. An example of the regression problem is the estimation of the property cost based on location, size, age, etc.

**Generation** problem defines a model that generates realistic but previously unseen samples from the data distribution. In general, these models operate without additional input, however, in recent years, guided generation–where parameters of the sample are given as input–became a developing research topic. The model should predict $y \in \mathcal{X}$ defined in such a way that every possible sample of the data also belongs to it $x \in \mathcal{X}; \forall x$, however, $y \notin \{x\}^N$.

However, each problem can be subdivided into many more levels. For example, text generation differs from text completion, which, in turn, differs from the problem of code completion. I will further focus on specific tasks related to the presented work.

### 1.2.2 Models

The key idea of machine learning is to replace the hand-crafted parameters for the data processing algorithms with the parameters derived directly from the available data. For example, instead of the manual selection of the brightness threshold to select all pixels belonging to bones in a CT volume, statistically deriving the proper threshold from several volumes with outlined bones could be employed. This solution can be viewed as a one-parameter (the threshold value) classification model solving the task of the segmentation (pixel-wise classification of an image). The object incorporating those derived parameters is called a model, and the process of the derivation of the parameters is called model training.

There are many very specialized model types, but I will resort to describing the most important for this work.

**k Nearest Neighbors** or k-NN is the simplest model, where (like in a look-up table) the parameters are the whole set of the training examples. When the new input comes, the decision is made as an interpolation between the labels of the $k$ nearest neighbors from the training examples. For the regression task, this is typically a linear interpolation; for the classification, it is majority voting. Different measures of closeness can be used.

**Linear Model** is a model where the weighted sum of the parameters of the input provides the prediction. The parameters of the model are, therefore, these weights. For the input sample $x$ of size $k$, the prediction is provided as $\hat{y} = \sum_{i=0}^{k} w_i * x_i + b$, hence the name. To solve the classification task with this model, the sigmoid function is added on top of the weighted sum, limiting predictions to the $[0, 1]$ range.

**Decision Tree** is a model represented by a binary tree, where each node contains a decision rule navigating each input to the left or right branches. Each leaf of this tree contains a prediction in the form of either a class for a classification problem or a value for a regression problem. Typically, the decision rule is a threshold (decision boundary) imposed on a specific parameter; e.g., `('weight', 80.1)`, would mean, that patients with weight more than 80.1 would be navigated to the right branch, and others to the left.

**(Artificial) Neural Network** or Deep Learning model is the most advanced type of model. This model can be seen as a series of linear models, with non-linear functions on top of each. Each linear model in this sequence is called a layer. There are multiple different layers designed specifically for the corresponding task, (e.g., convolutional layers designed for the image analysis, or recurrent layers developed for the text generation).

### 1.2.3 Supervision Types

For the training of the aforementioned models, one needs a dataset. Generally, all training types with respect to datasets could be divided into three: fully supervised, unsupervised, and semi-supervised.

**Full supervision** supposes that each example used for training has a corresponding label – the desired response of the model to this input. This is the most typical way of training a model, for example, ResNet models trained on the ImageNet

dataset or the YOLO model family which is trained on the COCO dataset are commonly used as a starting point for further training or as a readily available solution for some tasks. However, datasets with this type of supervision are costly to obtain with enough samples to saturate the model's capacity to train. And open datasets are rarely available for a specific task to solve, which is typical for medicine or biology.

**Unsupervised** training uses zero labels, only the input data, the model itself should implicitly contain our beliefs about the world, which results in a proper response. For example, the clusterization of samples, or training a generative model can be done without any supervision However, this type of training rarely is able to yield desired results itself and rather belongs to a class of exploratory analysis tools. There are specific classes of tasks, which are perfectly solved by such models though, e.g., anomaly detection (Zong et al., 2018).

**Semi-supervised** training is a relatively novel research area, boomed when the lack of labeled data became apparent. It includes such areas as self-supervised learning, self-training, weakly supervised learning, etc. Labels in this type aren't completely absent, however, they are either rough (hence, cheaper to produce), or provided only for part of the available examples. The aim, therefore, is to utilize the unlabeled data to train the model to produce meaningful representations of the input data. Further on, either a small model can be trained to utilize these representations for the actual task of interest, or the whole model could be fine-tuned to solve the actual (so-called downstream) task. The models trained in a semi-supervised way have great potential and are now a growing area of research, which can be seen by recent influential works of Oquab et al., 2023; Kirillov et al., 2023; OpenAI, 2023

## 1.3 Summary and Organization of the Thesis

In this introduction, we have traversed the evolution and application of Computed Tomography (CT), highlighting its pivotal role across such fields as medicine, biology, and material science. From its origins in basic observational techniques to the advanced, non-intrusive insights provided by CT, we've seen a transformative impact on scientific understanding and diagnostic processes. Particularly, the advent of CT has revolutionized the approach to visualizing internal structures, enabling precise disease diagnosis, understanding biological phenomena, and enhancing material analysis without destructive methods. Moreover, the integration of Machine Learning (ML) with CT data analysis was discussed, marking a significant leap in processing and interpreting complex data. These advancements exemplify the potential of combining deep learning with volumetric imaging, offering unprecedented accuracy and efficiency in data analysis. As we venture further, it's clear that the synergy between CT technology and ML not only broadens our investigative capabilities but also introduces challenges and opportunities in data handling, requiring innovative solutions to fully harness their combined power.

The rest of the thesis will be organised as a series of chapters consequently covering the main practical complications and their solutions in the same order they happen in the practical pipeline. In Chapter 3 I introduce my contributions to the first step of the pipeline – data pre-processing. Namely, noise removal. In Chapter 4 the optimization of the pipeline of the labeling is presented. The idea here is to minimize expert labour on labeling, while maximizing the performance of the model

trained on such labels. In Chapter 5 I further propose a way, to pre-train a model on an unlabeled dataset. As was mentioned earlier, pre-training helps to initialize the model used for further analysis, with weights that are closer to global minimum, than the random initialization. In Chapter 6 the final contribution is presented – a self-training algorithm. The self-training here denotes a technique that allows tuning an already trained a model with additional data without requirement to have it labeled.

# Chapter 2

# Related Works

In this chapter, I will quickly recap the information about the whole data acquisition and processing pipeline and outline bottlenecks that are open to being solved with machine learning. Then, I will describe the state of the art in solving these problems.

## 2.1 Sample Analysis Bottlenecks

The whole process of sample analysis, which we want to speed up, from acquisition to conclusions drawing, can be roughly separated into three main steps.

**Sample preparation** includes all biological, chemical, or engineering phases required to obtain a sample. It will not be covered in this work, since this process is mostly driven by biology and chemistry experts, and wildly varies depending on the sample, imaging device, and aims of the research. See (Weinhardt et al., 2018) as an example.

**Sample imaging** , or measurement, is the process of obtaining an image of the prepared sample. This work is limited to 3D CT images, radiography, and spectrography. However, generally, this could include text, sound, or photographic images.

**Image analysis** includes manual analysis, labeling datasets, training and inference of predictive models, statistical analysis, hypotheses testing, and all other processes required to draw scientific conclusions about the sample under investigation.

This work is focused on sample imaging and analysis, and further in this section, I will describe their corresponding details.

### 2.1.1 Sample Imaging

Modern-day measurement devices (including synchrotron facilities) aim to provide as much automation of the measurement as possible. It includes robotic sample operators (Kamp et al., 2018), automated procedures of beam alignment (Campbell et al., 2021), fast automated post-processing (Faragó et al., 2022), and a neverending race to increase the capacity of data transfer channels.

All these, multiplied by the increasing amount of facilities and the trend to open access to scientific data, increases the amount of data available for analysis. Generally, dataset size in MRI and CT areas grows exponentially, as it is noted by (Kiryati and Landau, 2021). As an example for KARA facility, for the project of morphometrical analysis of the Medaka fish, more than 850 samples were scanned and analyzed, with an average image taking 100Gb of disk space, without markup.

Yet, due to the limitations of the detectors and stochasticity of the particle counting process, there is a bottleneck in exposure time. The first is a physical limitation, sources aren't able to produce a high flux of particles of any given energy. The energies of the particles emitted by the source are distributed in a way resembling Normal distribution. An example of such distribution is shown Figure 3.1. Therefore, moving away from the peak flux energy requires increasing the exposure time to keep the PSNR same. Other possible sources of quality limitation are *in-vivo* or *operando* imaging. In the *in-vivo* case, the sample is a living creature and has a limited lifetime under exposure to radiation. In both *in-vivo* and *operando* cases, the speed of the process could also limit the exposure time, since imaging with increased exposure produces blurry images and fails to resolve the process of interest in the time domain. To compensate for this, denoising techniques are employed. I will describe my contribution to solving the denoising task in 3.

### 2.1.2 Image Analysis

As the amount of information grows, the natural problem is to increase the speed and ease of processing. A multitude of tools is developed to help area experts with their work of analysis. Some aimed to help segment the volumetric data (Lösel et al., 2020), others aimed to help visualization and further analysis (Wolf et al., 2004). The most popular way however is the data-driven automation of the analysis based on the machine learning approaches. As mentioned before, DL models overall tend to be eager for training data. While the amount of the data itself rapidly grows, the amount of readily available markup for an arbitrary task is almost always a zero. The next bottleneck, therefore, is the amount of time the area experts need to contribute to labeling the datasets for further training. I describe my contributions towards optimizing the labeling procedure for the segmentation of CT volumes in Chapter 4. Another way to mitigate this is to use semi-supervised learning approaches. In Chapters 5 and 6, I describe my contribution towards using self-supervised learning and self-training to make use of the unlabeled data.

## 2.2 State of The Art

In this section, I will describe the techniques, that will be used in this work, in a more detailed fashion. First, in Section 2.2.1, I will cover the Deep Learning basics required for this work. In Section 2.2.2, I will describe methods used to solve the task of image denoising. Finally, self-supervised pre-training and self-training methods will be described in Section 2.2.2 and Section 2.2.2 respectively.

### 2.2.1 Models

As it was mentioned before, modern-day neural networks, are a sequence of layers, where each layer is some linear transformation of the input with some non-linear function appended. The combination of characteristics of these layers (dimensionality, the non-linear function used, where the input comes from) is called the model architecture. And the numerical parameters of the linear transformations are called weights. The model altogether is typically denoted as a function $f_\theta$, parametrized with weights $\theta$. Typically, the architecture of the model is fixed by a researcher. There's one exclusive area of research (architecture search algorithms) where the architecture is optimized for the specific task automatically, but this is out of the scope of the current work. The training of the model is, therefore, the optimization of those

weights $\theta$. Here I will describe how the training of the neural networks is conducted, and, later on, specific layers and architectures important for computer vision.

### Training

The training of the Deep Learning model requires three principal components: a dataset, a loss function, and an optimization algorithm.

**The data set** is a subsample of the general data distribution with corresponding labels attached to each sample. Consider some general data distribution $\mathcal{X}$. For example, it could be the distribution of all possible chest MRI scans, then one data point $x_i \in \mathcal{X}$ is one volume. The attached label $y_i$ depends on the task, it could be segmentation of the lungs or some other organs, diagnosis of the patient, or even the binary label of being ill with some specific disease. The data set then, is a collection of $N$ sample-label pairs $\{(x_i, y_i)\}_(i = 0)^N$

Generally, the label is provided by an expert, for example by manually segmenting the organs of interest. For self-supervised learning, however, the label is inferred from the available data. E.g., for the colorful image re-colorization (Zhang, Isola, and Efros, 2016), the dataset is collected as $\{to\_greyscale(x_i), x_i\}_{i=0}^N$, where $x_i$ is drawn from the full distribution of the colorful photos.

The ideal target of training is to find such optimal weights $\theta^*$, that $f_{\theta^*}(x_i)$ is as close to $y_i$ for any $x_i \in \mathcal{X}$ as possible. However, since our dataset is limited by $N$ samples, the best we can do is to fit a model that performs well on this representative subsample, which is called Empirical Risk Minimization.

**Loss Function** also called cost function, takes as input the prediction of the model and corresponding label $(f_\theta(x_i), y_i)$ and maps it to $\mathcal{R}$. The loss function (typically denoted as $\mathcal{L}$) is used to estimate the badness of the prediction. So, if for two models $f_{\theta_1}, f_{\theta_2}$, it's true that $\mathcal{L}(f_{\theta_1}(x_i), y_i) < \mathcal{L}(f_{\theta_2}(x_i), y_i)$, we will say that the model $f_{\theta_1}$ has a lower loss, and therefore performs better for the $i$-th sample. The aim of the model training, therefore, can be reformulated as $\sum_i \mathcal{L}(f_\theta(x_i), y_i) \underset{\theta}{\rightarrow} min$.

The most common examples of the loss functions for the regression task are L1-loss $\mathcal{L}(y', y) := |y' - y|$, or L2-loss $\mathcal{L}(y', y) := |y' - y|_2^2$. For the classification task, the most popular option is so-called cross-entropy loss $\mathcal{L}(y', y) := \sum_c y_c \log y'_c$ , where $c$ is the index of the class in the multiclass classification problem.

**The Optimization Algorithm** therefore, should minimize the loss function value over the dataset, w.r.t. the weights. It is commonly accepted, that for better optimization access to the gradient is preferred. To make the gradient available, the chain rule is used. Chain rule, states, that if we have a function, expressed as a sequence of two functions $d(x) := f(g(x))$, then its derivative can be expressed as

$$\frac{\partial d(x)}{\partial x} = \frac{\partial f(g(x))}{\partial x} = \frac{\partial f(g(x))}{\partial g(x)} \frac{\partial g(x)}{\partial x}$$

This rule, being applied sequentially to each layer starting from the last allows to calculate, is called Backward Propagation.

With the gradient available, the most common algorithm of the optimization is the Stochastic Gradient Descent. The normal gradient descent updates weights with

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta_t} (\sum_i \mathcal{L}(f_{\theta_t}(x_i), y_i))$$

, where $\eta$ is a small constant, called the learning rate. But since the computation of the gradients for the whole dataset is problematic, SGD uses batches of the inputs to update the weights. For the sake of simplicity, I will use $\mathcal{L}_{\theta_t}$ to denote a loss, calculated for the weights $\theta_t$ on one batch of the data. The SGD update then is

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta_t} \mathcal{L}_{\theta_t}$$

Each update then is called a step, and when the model was updated on the whole dataset (so, enough batches to see each $x_i$ in the dataset), it's called an epoch.

There is a multitude of algorithms varying the optimization method. However, the most common optimizer nowadays is Adam (at the moment I write this, the paper had almost 138000 citations), which combined ideas from the RMSProp and Momentum. The idea of the Momentum update is to overcome small local minima by updating the weights with aggregation of the last gradients, instead of the only very last. This way, the model is expected to move along the smooth slope of the general plane, instead of bouncing around from every small change in the slope. The idea of the RMSProp was to alleviate problems of the low slope along some parameter axes, which can be triggered, for example, by the rarely occurring inputs. This method dynamically scales the updates along different axes, based on their aggregated magnitude. Overall, the update by the Adam optimizer updates are described in Equation (2.1). Where, $m$ denotes momentum, and $v$ normalization coefficients; $\beta$ and $\gamma$ are the smoothing factors for the exponential smoothing of the momentum and normalization; $\tilde{m}$ and $\tilde{v}$ are momentum and normalization coefficients after corrections to remove skewness of the estimation towards 0, introduced by initialization of the estimators.

$$\begin{aligned} m_{t+1} &= \beta m_t + (1-\beta) \nabla_{\theta_t} \mathcal{L}_{\theta_t} \\ v_{t+1} &= \gamma v_t + (1-\gamma)(\nabla_{\theta_t} \mathcal{L}_{\theta_t})^2 \\ \tilde{m}_{t+1} &= \frac{m_{t+1}}{1-\beta^t} \\ \tilde{v}_{t+1} &= \frac{v_{t+1}}{1-\gamma^t} \\ \theta_{t+1} &= \theta_t - \eta \frac{\tilde{m}_{t+1}}{\sqrt{\tilde{v_{t+1}}} + \epsilon} \end{aligned} \tag{2.1}$$

**Layers**

In a Neural Network, a layer is a collection of nodes or neurons that process the inputs received from the previous layer and produce outputs for the next layer. Each neuron in a layer is connected to all the neurons in the previous layer, and each connection has a weight that determines the strength of the signal transmitted through it. In this section, I will list the layers, that are most important to understand the current work.

Initially, one layer of the neural network was a linear transformation, appended with a non-linear function. This model was dictated by a biological analogy of a firing neuron employed by the McCulloch and Pitts, 1943. This layer is called a **dense layer** in modern literature and is calculated as $y = xA^T + b$, where $A$ is a matrix of shape `input_size` $\times$ `output_size`, $b$ is a vector of length `output_size`, $x$ is the input vector of size `input_size`, and $y$ is the output vector of size `output_size`. The so-called nonlinearity applied on top of the linear transformation was, initially, mostly the logistic function $f(x) = \frac{1}{1+\exp x}$, as it is a differentiable approximation of the step function dictated by the biological analogy. Nowadays, there is a lot of research on the activation functions, but the most common function nowadays is the Rectified Linear Unit, or ReLU, defined as $f(x) = \max(0, x)$.

Next in importance for this work is the **convolutional layer** introduced by Lecun et al., 1998. Convolutional layers are a type of layer in a Neural Network that are specifically designed for processing data with a close relation between adjacent inputs, typically applied to images. The idea is based on kernel filters, a popular concept in computer vision; e.g., the Sobel filter is used for edge detection and the Gaussian filter for denoising. The convolutional layer consists of several such filters with the same kernel size, however, the parameters (weights) of the kernels are fitted together with the whole model to minimize the loss. The output from each filter is called a channel, and all channels are stacked together upon computation. The output of a $c$-th channel can be calculated as $y_c = A_c \star x + b_c$. Where $\star$ is the operation of the convolution, $A_c$ is a tensor of size `input_channels` $\times$ `kernel_size` $\times$ `kernel_size` which contains weights of one kernel filter for the $c$-th channel, $b_c$ is the bias for the $c$-th channel, $x$ is the whole input of the layer, the image of size `input_channels` $\times$ `input_width` $\times$ `input_height`, and the $y_c$ is the output of the $c$-th filter of size 1 $\times$ `output_width` $\times$ `output_height`. The $c$ here is used to index along the filters of the layer, the first dimension of the full output will be dependent on the number of filters used. The `output_width` can be calculated as `input_width` - `kernel_size` + 1.

The third important layer for **computer vision** models is the pooling layer. The pooling layer operates on each channel of its input and statistic of the nearby outputs. A hyper-parameter of the pooling layer is the size of the neighborhood to be aggregated. For example, it could be $2 \times 2$ neighborhood, therefore producing one output pixel per field of $2 \times 2$ input pixels. An extreme example of it is called Global Pooling – when the neighborhood to aggregate is the whole input. The two most common types of pooling layers are max pooling and average pooling. These layers help to aggregate data from large images, effectively discarding non-important signals. The global pooling layers allow models to be independent of the input image shape.

The next important layer is the **dropout layer**. It is a layer, primarily used as a regularization technique to prevent overfitting. It works differently during training and inference. During the training phase, it randomly sets a portion of the incoming activations to zero, before passing it to the next layer. That forces the network to learn more robust and generalizable features, instead of being overly reliant on one particular activation. The portion of activations, which are to be set to zero is a hyper-parameter of the layer and is called the *dropout rate*. During the inference phase, however, all the inputs are passed to the next layer. But to avoid a shift in the average input values, all activations are multiplied by a dropout rate. In convolutional layers, there is more freedom to what to denote as one activation. The most typically used is the spatial dropout, which equalizes to zero random "pixels" of the embedding map fed to it as an input. However, there are works considering channel

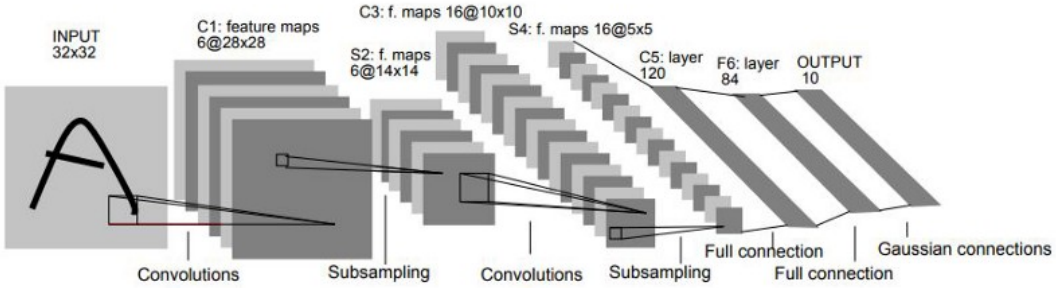



FIGURE 2.1: The architecture of the LeNet-5 model, as presented in (Lecun et al., 1998)

dropout, where the whole channel is equalized to zero, or other, more exotic types of dropout.

There are many more layers less important for this work, e.g., the self-attention layer introduced in transformers, or the recurrent cell layer, introduced by RNN. Since they aren't that important for this work, I leave a thorough walkthrough of all layers to other sources.

### Architectures

Having layers–the building blocks of neural networks–defined, I will move on and describe the most important architectures, that will be used in this work. For this work, it is essential to understand models designed for two tasks: classification and segmentation. For each of those tasks, I will describe two models, one which sparked the progress in the current direction, and one which is the most popular one, typically used throughout this work.

LeNet-5 is a pioneering convolutional neural network architecture developed by Lecun et al., 1998 which revolutionized the neural networks for computed vision by introducing the convolutional layers. LeNet-5 consists of two sets of convolutional and pooling layers, followed by three fully connected layers Figure 2.1. The fully connected layer was enabled by the fact, that the images in the target dataset were all of the same size ($28 \times 28$ px). Therefore, after the set of the downscaling operations, the incoming image always became a representation of size $C \times 1 \times 1$, where $C$ is the number of channels. The network is relatively shallow compared to modern architectures, but it introduced many of the concepts that are still used in modern CNNs. However, stacking more layers was problematic due to the vanishing gradient problem (Simonyan and Zisserman, 2015).

The problem of vanishing gradients was solved by the ResNet model. To enable the training of very deep networks authors utilized residual connections. The model consisted of the so-called residual blocks, each containing several convolution and pooling layers. However, each residual layer has a bypass connection directly passing its input to the next block, which allows easier gradient propagation to deeper blocks without vanishing Figure 2.2. To enable training with input images of any size, the authors adapted max-pooling by Ranzato et al., 2007 and changed it to a global average pooling layer between the convolutional layers and the fully connected layer. This layer helped to preserve global features from the whole image context, without making the model dependent on the image size. The ResNet

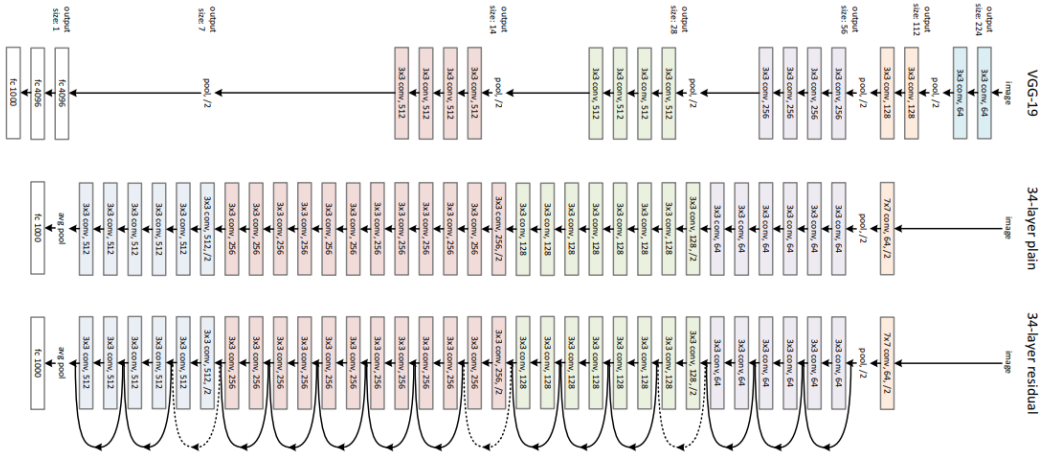



FIGURE 2.2: The architecture comparison of the ResNet to VGG and plain architecture of the same size. As presented in (He et al., 2016a)

is nowadays amongst the most used architecture of CNNs and will be widely employed in this work. For a deeper understanding of CNNs' history and architecture, I refer a reader to (Khan et al., 2020).

However, while the ResNet is a good architecture for classification, it has no use for segmentation, since it predicts only one label per image, while segmentation requires the prediction of one label per pixel of the input image. The Fully Convolutional Network (FCN) is a type of neural network architecture that is specifically designed for semantic segmentation tasks. Unlike traditional CNNs, which are designed for classification tasks, FCNs use only convolutional layers and do not have any fully connected layers. The simplest way to construct an FCN from a ResNet requires (1) replacing the Global Average Pooling layer with an upscaling layer, and (2) replacing the fully connected layer with a convolution layer. The upscaling layer parameters are calculated from the parameters of previous layers of the model in such a way, that it upscales the image to its original size. The convolutional layers are of size $1 \times 1$, which makes them equivalent to the operation of applying the dense classification layer to each pixel separately. This way, even a model trained to be a classifier, can be converted to present a rough segmentation mask.

U-Net is a convolutional neural network architecture designed for biomedical image segmentation tasks. The U-Net somewhat builds on the idea of the fully convolutional model. However, it (1) uses information from several different layers of the so-called encoder model (e.g., ResNet), and (2) replaces simple upscaling operations with upscale-convolve layer pairs, which allows smart upscaling that modifies the output progressively during upscaling. The U-Net is one of the most used models for image segmentation. One particular benefit of this model is that the architecture allows using a pre-trained encoder while training the decoder part of the model, this technique is widely used, and good pre-trained encoders are publicly available.

### 2.2.2 Tasks

Now that the elements of the Neural Networks for computer vision are introduced, I will make a deeper introduction to the tasks, which are considered in this work. Namely, the denoising task – the task of the noise removal from the images, the pretraining tasks that aim to find such data-driven initialization of the model, that it

has better properties (e.g., convergence speed or better minimum achieved), and the self-training technique, that utilizes the ability of a model trained on a small set of data, to improve itself while observing the larger set of the unlabeled data.

### Denoising

The denoising task with stochastic artifacts caused by image acquisition. These artifacts are inherent to digitally acquired images, as an acquisition process (also known as acquisition function) is subject to many uncertainties and in general, is not accurately known. The acquisition function includes optical distortion, lens and detector array heterogeneity, and inherent noise driven by the stochastic nature of the particle emission and their interaction with materials. As optical distortion is a simple misplacement of information, a distortion map can be estimated and applied to compensate for it. For the lens and matrix heterogeneity, flat and dark field corrections are used. Flat and dark fields refer to the measurement of detector response with and without source illumination, respectively. Finally, denoising is used to compensate for the inherent stochastic noise. Hence, the denoising problem is to restore signal $S$ from a noisy observation $I$ (Gonzalez and Woods, 2008):

$$I = S + \sigma(S) \tag{2.2}$$

where $\sigma(S)$ is the inherent noise of the imaging device.

The most basic denoizing methods are spatial filtering methods: mean, median, or Gaussian kernel filters (Gonzalez and Woods, 2008). For each pixel, these filters select a new value, based on the weighted values of the neighboring pixels. These filters are fast, robust, well-understood, and work fairly well in many situations. The main drawback of these classical filters is their tendency to not only remove the bright noise outliers but also blur the sharp edges.

More advanced spatial filtering approaches, *e.g.*, non-local means (NLM), use more information from the whole image (Buades, Coll, and Morel, 2005). Instead of taking an average of the direct neighborhood of the pixel, NLM takes an average of the whole image, weighted by similarity and distance between the "donor" and the "recipient" pixel. The process of revisiting multiple locations in the image, comparing their surroundings, and computing the average can take minutes for one image. In return, this method is capable of producing sharper denoised images (Fan et al., 2019).

Alternatively, denoising can be formulated as an optimization problem and regularization can be used to incorporate some prior knowledge about the image properties (Gu and Timofte, 2019). These methods are very powerful but require deep mathematical knowledge and handcrafted regularizers, making their application challenging. One of the most successful regularizers is Total Variation (TV) which encourages piece-wise constant image regions separated with sharp boundaries (Rodríguez, 2013).

To summarize, classical methods require fine parameter tuning by an expert to balance smoothing and denoising, hence there is a significant risk of information loss if applied incorrectly. A more comprehensive overview of the classical denoising methods can be found elsewhere (Fan et al., 2019). I describe my contributions towards the denoising in Chapter 3

### Self-Supervised Pre-Training

Most part of this work is dedicated to self-supervised methods, I will introduce them more in-depth in the following paragraphs. The idea behind self-supervised learning is to create a fully-supervised task from data without labels. This task (so-called pretext), doesn't directly lead to a model that is able to solve the actual task. However, training for a pretext task forces the model to create representations beneficial for the actual (so-called downstream) task solving. A great example of such a task is image colorization. On one hand, having just a set of colorful images, it's easy to produce a dataset consisting of grayscale photos paired with their colorful version. On the other hand, the conversion of the grayscale images back to full color requires a model to develop a deep understanding of the depicted structures to make a prediction. As a result, the model trained to colorize the images is forced to recognize the depicted objects and requires less markup to be trained to segment them (Larsson, Maire, and Shakhnarovich, 2017).

Two major directions of self-supervised machine learning are knowledge-prior and data-prior training.

**Knowledge-prior training** uses expert knowledge of the specific task for training. For example, as a pretext task, a model was trained to predict the distance between different splits of the brain cortex (Spitzer et al., 2018). As a downstream task, the segmentation of the cortex into specific regions was used. To train this model properly, the authors knew a specific way to measure the distance between those slices, because they knew the way the cortex is organized.

**Data-prior training** in contrast isn't bound to a specific data type. The most popular subtype is contrastive learning. In the contrastive learning framework for each image in a batch, two copies are generated, and they are passed through different augmentations. The loss enforces that representation of the different views of the same image should be close to each other, while representations of the different images should be distant. While the model is forced to recognize objects on the image to judge their closeness beyond simple features, destroyed by augmentations.

The benefits of data-driven self-supervised training are the ability to be applied to almost any dataset and the broad set of studies for these approaches since they are generally applicable. However, knowledge-driven approaches can provide better results by being tailored for specific tasks and datasets.

**Contrastive Learning** Recently, a lot of research has focused on the so-called *contrastive learning* (Newell and Deng, 2020). Contrastive SSL tasks are designed to distinguish between positive and negative examples. In (Chen et al., 2020a) authors presented SimCLR – a method for contrastive learning, which quickly became a popular approach. It relies on a specific procedure of batch construction – a number of samples are taken from a dataset, then two *views* of each image are created by applying random augmentations. The model is trained to pull together embeddings of different views of the same image (positive samples) and push away from embeddings of different images (negative samples). As a downside, this approach requires a large batch size containing many negative samples. This leads to a higher memory footprint and longer computation time. Another difficulty for SimCLR is the requirement that the positive and the negative samples should be as diverse as possible. Currently, many methods aim to tackle this by imposing constraints on the

embedded vectors themselves (Bardes, Ponce, and LeCun, 2021). Moreover, it was noted that SimCLR might be sensitive to harder augmentations (Wang et al., 2021), requiring fine-tuning of the method. More details on various contrastive methods are given in the review by Weng, 2021.

**Knowledge-Driven Learning** Knowledge-Driven methods are based on the specific knowledge of an application area and dataset. In the work, by Spitzer et al., 2018 the authors propose to employ the geodesic distance between two patches of the human brain cortex. Using geodesic distance was an important metric due to the inherent 3D structure of a brain cortex – its characteristic folds with high curvature. In another work (Haghighi et al., 2021), the authors proposed to rely on the fact that many medical images are highly structured (i.e., relative positions between organs) and aligned during a scanning procedure (e.g., radiography image of lungs). Thus, they supposed that taking crops from the same location of an image would result in the same structures being demonstrated on the crop. The authors, therefore, selected several pre-defined crop locations and assigned a separate class for each of those locations. These classes were called Transferable Visual Words. It is important to note, that the strong assumption about the alignment of data requires thorough data filtering, even for medical applications. I describe my contributions towards the self-supervised pre-training in Chapter 5.

### Self-Training

Another training type used in this work is self-training. The self-training technique, in contrast to self-supervised training, employs the unlabelled part of the dataset directly to train a model for the target task (Hsu et al., 2019). To do so, two models are used, the Teacher and the Student. The Teacher model (which could be an ensemble of models) is trained on a labeled part of the dataset. Subsequently, its predictions on the unlabelled part of the dataset (so-called pseudo-labels) are used to train the Student model. This method originates from the Knowledge Distillation framework (Hinton, Vinyals, and Dean, 2015). Experiments show, that given enough data available, self-training may be superior to the pre-training (Zoph et al., 2020). In that work, the authors also show that self-training could be beneficially combined with self-supervised pre-training.

Several methods were proposed to improve the results of the self-training further. Authors proposed to train several Teacher models and average their predictions to form pseudo-labels (Tarvainen and Valpola, 2017). Another work proposed to use soft labeling instead of hard labeling. It is, instead of selecting the most probable class, the authors proposed to save the whole probability distribution for each prediction. This required updating the loss, and making use of KL-divergence instead of the Cross-Entropy for prediction, but was shown to be beneficial for the final quality (Xie et al., 2020). One more work proposed a procedure to select the best pseudo-labels for the Student model training based on the confidence of the Teacher's predictions (Zou et al., 2018a). My contributions towards the self-training in Chapter 6

# Chapter 3

# Self-Supervised Multi-Channel Data Denoising

The work presented in this chapter is strongly related to the (Zharov et al., 2023), which was published in the Optics Express Journal.
The work was performed in collaboration with *Dr. Evelina Ametova*. She simulated the computed tomography dataset and implemented the material decomposition algorithm. She also significantly contributed to the manuscript preparation.

As soon as the images land on a hard drive, and even before any automated processing takes place, they can be reviewed by a scientist. What unites both people and classical computer vision algorithms is the high noise sensitivity. Noisiness of an image can be measured with Signal-to-Noise Ration (SNR) defined as $SNR = \frac{\mu}{\sigma}$, where $\mu$ is the average value of some image neighbourhood and $\sigma$ its variance (Gonzalez and Woods, 2008). This chapter engages with the noise reduction (SNR increase) for a specific subset of tomographic and radiographic images, those with multiple channels.

Noise can become a particularly critical issue in applications where additional constraints force a strong reduction of the SNR per image. These constraints may result from limitations on the maximum available flux or permissible dose and the associated restriction on exposure time. Often, a high SNR per image is traded for the ability to distribute a given total exposure capacity per pixel over multiple channels, thus obtaining additional information about the object by the same total exposure time. These can be energy channels in the case of spectroscopic imaging or time channels in the case of time-resolved imaging. Conventional image denoising methods work on a per-image basis and rely on certain assumptions concerning image properties. Consequently, they perform well when the assumptions are met and fail otherwise. At the same time, tremendous progress in machine learning demonstrated that data-driven methods are much more flexible in accommodating various image characteristics.

The proposed method is designed specifically for the multichannel (time or energy-resolved) imaging datasets and relies on the recent *Noise2Noise* (N2N) (Lehtinen et

al., 2018) self-supervised denoising approach. that learns to predict a noise-free signal without access to noise-free data. N2N in turn requires drawing pairs of samples from a data distribution sharing identical signals while being exposed to different samples of random noise. The proposed method is applicable if adjacent channels share enough information to provide images with similar enough information but independent noise. I demonstrate several representative case studies, namely spectroscopic (k-edge) computed tomography, *in vivo* cine-radiography, and energy-dispersive (Bragg edge) neutron tomography. In all cases, the N2N method shows dramatic improvement and outperforms conventional denoising methods. For such imaging techniques, the method can therefore significantly improve image quality, or maintain image quality with further reduced exposure time per image.

## 3.1 Introduction

Many imaging modalities rely on radiation's penetration ability to make an object's interior visible. Physical interactions of radiation and matter, such as absorption, scattering, or phase shifts, can be used to obtain contrast inside the object of interest. Commonly, either *radiography* (single view projection) or *tomography* (multiple views with subsequent volumetric image reconstruction) are acquired. As particle or photon emission and detection are stochastic processes, and often source flux and detector efficiency are limited, longer exposure time improves image quality. However, there are many scenarios where sufficient exposure can not be achieved. An obvious example is *in vivo* imaging, where the radiation dose ultimately limits the amount of information acquired (Moosmann et al., 2013). Another example is spectroscopic imaging with a polychromatic beam, where the detected intensity or particle counts are distributed across multiple energy bins. This leads to a significant noise per energy channel or requires a dramatic increase of exposure times, hence, limiting the experiment throughput (Warr et al., 2021). Both imaging modes can be generalized as *multi-channel* images. In this chapter, I specifically address a number of cases when the channels of multi-channel images share a sufficient amount of common structural information but are affected by independent noise samples.

Given the aforementioned physical constraints, we often need to rely on image processing techniques to improve image quality and extract valuable data. A group of methods for improving image quality that is affected by noise is called *denoising*. Similar to other domains, methods based on *Machine Learning* (ML) have revolutionized denoising (Ilesanmi and Ilesanmi, 2021). In my work, I demonstrate an ML approach to improve the quality of underexposed images in challenging applications such as spectroscopic k-edge computed tomography, *in vivo* radiography, and energy-dispersive Bragg-edge neutron tomography. The method is based on the recent Noise2Noise (N2N) self-supervised denoising approach (Lehtinen et al., 2018). The main assumption enabling the N2N method is formulated as follows: Consider two images $I_1$ and $I_2$ that share the same structural information $S$ but are affected by independent and identically distributed (iid) instances of noise $\sigma_1$ and $\sigma_2$. If the model is trained to predict $I_1$, given $I_2$ as input, the best prediction possible is $S$, because $\sigma_2$ is conditionally independent of $\sigma_1$, given $S$.

This chapter is organized as follows. First, the related work is outlined to put the proposed method in context. Then, the N2N method is described, followed by three rigorous case studies. Finally, a discussion of the findings in the context of multi-channel imaging is provided.

## 3.2 Related Work

Artifacts are inherent to digitally acquired images, as an acquisition function is subject to many uncertainties and is generally not accurately known. The acquisition function includes source or detector heterogeneity, optical distortion by optical elements or diffraction during wave-field propagation, and inherent noise driven by the stochastic nature of particle emission, detection, and interactions. As optical distortion is a misplacement of information, a distortion map can be estimated and applied to compensate for it. For a number of cases, simple flat and dark field corrections are applicable. Flat and dark fields refer to the measurement of detector response with and without source illumination, respectively. Finally, denoising is used to compensate for the inherent stochastic noise. Hence, the denoising problem, as it was introduced in the Chapter 1, is to restore the (deterministic) signal $S$ from a noisy observation $I$:

$$I = S + \sigma(S) \tag{3.1}$$

where $\sigma(S)$ is the inherent noise of the imaging device. This noise depends not only on the stochasticity of the particles (neutrons, photons, etc.) and the electronics but additionally on the distortions, and also on the transformation applied to correct the image. All this makes a closed-form distribution estimation problematic.

The existing image denoising approaches can be roughly categorized into two large groups: classical image processing and ML approaches. Typically, classical image processing approaches work in a single-image manner and incorporate expert beliefs about the nature of noise. ML approaches, on the other hand, employ the idea of fitting a data-driven model entirely without or with minimal expert knowledge about the nature of the data.

### 3.2.1 Classical Image Processing

The basic spatial filtering methods are mean, median, or Gaussian kernel filters (Gonzalez and Woods, 2008). For each pixel, these filters select a new value, based on the weighted values of the neighboring pixels. These filters are fast, robust, well-understood, and work fairly well in many situations. The main drawback of these classical filters is their tendency to blur sharp edges.

More advanced spatial filtering approaches, e.g., non-local means (NLM), use more information from the whole image (Buades, Coll, and Morel, 2005). Instead of taking an average of the direct neighbourhood of the pixel, NLM takes an average of the large region, weighted by the similarity between the "donor" and the "recipient" pixel. The process of revisiting multiple locations in the image, comparing their surroundings, and computing the average can take minutes for one image. In return, this method is capable of producing sharper denoised images (Fan et al., 2019).

Alternatively, denoising can be formulated as an optimization problem and regularization can be used to incorporate some prior knowledge about the image properties (Gu and Timofte, 2019). These methods are very powerful but require deep mathematical knowledge and handcrafted regularizers in many cases, making their application for experimental data challenging. One of the most successful regularizers is Total Variation (TV) which encourages piece-wise constant image regions with sharp boundaries (Rodríguez, 2013). In summary, classical methods require fine tuning of parameters by an expert to balance smoothing and denoising. Hence there is a significant risk of information loss if applied incorrectly. A more comprehensive overview of classical denoising methods can be found elsewhere (Fan et al., 2019).

### 3.2.2 Machine Learning Approaches

The evolution of classical methods may be seen as a series of steps taken to increase the amount of information used to correct a single pixel value. In this respect, ML-based approaches appear as a natural further step: a *model*, trained to correct the noise, implicitly incorporates knowledge about the whole dataset.

Early ML-based image denoising approaches worked in a supervised manner, i.e. a model was trained on a set of noisy images to predict a noise-free image (*target*). Recently, authors of the N2N method demonstrated that there is no need for a noise-free target: if one uses a pair of noisy images (affected by iid instances of the noise) as an input and as a target for the training, the model will predict the noise-free image (Lehtinen et al., 2018). The underlying intuition is that independent instances of noise are uncorrelated and cannot be predicted, hence the model is forced to extract features. Even though N2N does not explicitly require a set of noise-free images, the Lehtinen et al., 2018 synthetically formed noisy pairs by adding noise to noise-free images.

There have been several attempts to extend the N2N method for denoising problems where pairs of images are not naturally available. Noise2Self (Batson and Royer, 2019) and Noise2Void (Krull, Buchholz, and Jug, 2019) generate the required pair of images by taking random pixels in the noisy image and disturbing them with yet another noise distribution. In this way, multiple training pairs can be constructed from a single noisy image. Noise2Stack (Papkov et al., 2021) was designed for three-dimensional tomographic data and is based on an assumption that tomographic data is typically smooth. Therefore, slice-to-slice changes are assumed to be significantly smaller than the slice-wise variability caused by noise, hence, neighbouring slices can be used for training.

Alternatively, constrained autoencoders can be used to denoise images (Vincent et al., 2008). During the training, autoencoders use the same image both as input and target and attempt to compress (encode) the input image into its lower-dimensional representation. The denoising properties of this approach rely on the assumption that the noise, due to its stochastic nature, is harder to encode, than the signal. To additionally limit the capacity of the model to store information about the noise, it can be restricted by limiting the computational capacity of the model, lowering the dimensionality of the learned representation, or introducing synthetic noise into it (Vincent et al., 2008). However, the autoencoders are inefficient if the noise is spatially correlated and can be easily memorized by the model. The Hierarchical DivNoising (HDN) method addresses this issue by training a variational autoencoder with a noise model imposed over output (Prakash et al., 2022). The authors proposed a way to find particular components of the model that encode information about the noise so that they can remove those components. Even though the proposed methods provide a valuable alternative to the N2N approach, the authors highlight that the N2N approach is a hard-to-beat baseline (Prakash et al., 2022).

## 3.3 Model Training

The N2N method assumes that a pair of images contains the same signal and iid noise. My adaptation of the method to multi-channel image data takes its inspiration from the denoising of Synthetic Aperture Radar (SAR) images (Dalsasso, Denis, and Tupin, 2022). In SAR imaging, both the phase and amplitude of received microwaves are measured in each pixel; commonly the phase information is ignored.

However, the authors demonstrated that the amplitude and the phase contain complementary information and can be used as a basis for N2N denoising. I hypothesize that in multi-channel imaging, adjacent time frames or energy levels indeed share sufficiently similar signals, and have noise samples close to being iid. Therefore, I generate the required pair images based on this hypothesis. To help the model catch complex spatial structures of the signal, I also feed it with multiple adjacent energies or time frames as input whenever it does not result in oversmoothing.

Following (Krull, Buchholz, and Jug, 2019; Batson and Royer, 2019; Lehtinen et al., 2018), I use the fully-convolutional neural networks as a model architecture. I employed U-Net with ResNet-50 (as implemented in (Pavel Iakubovskii, 2019)) as the backbone and relied on the Adam optimizer with a $3 \times 10^{-4}$ learning rate, without scheduling. Referring to the model, I will use $f_\theta$ and model interchangeably, where $\theta$ denotes trainable parameters of the neural network. The training, therefore, is the process of minimization of the proposed loss function (defined for specific experiments) by changing the parameters $\theta$. Each image pair was augmented with random crops, shifts, scale, rotations, distortions, and different types of blur. I acknowledge that there is room for quality improvement via larger models, modern architectures, better optimization procedures, or more aggressive augmentations. The sensitivity study of training parameters is a topic of future investigation.

## 3.4 Experiments

### 3.4.1 Simulated spectral X-ray tomography

As a first case study I will discuss the applicability of N2N to energy-dispersive X-ray tomography, which is of interest for biomedical imaging (Warr et al., 2021). The polychromatic emission of laboratory X-ray tube sources is suitable to provide sufficient photon flux. However, the broadband spectrum also leads to disadvantages in quantitative analysis. In the conventional absorption mode, each detector pixel integrates all photons irrespective of their energy. Since attenuation is a function of photon energy, conventional tomographic reconstruction might exhibit so-called beam-hardening artifacts (Davis, Jain, and Elliott, 2008). However, acquisition with an energy-dispersive X-ray detector allows segmenting materials that can be inseparable in polychromatic absorption contrast. These are materials with similar mean polychromatic absorption, but with spectrum showing sharp discontinuities at energies equal to the binding energies of the core-electron states, so-called *absorption-edges (K, L, M) edges*. Thie energy spectrum in each reconstructed voxel can be used to identify the corresponding material. Highly energy-dispersive (so-called hyperspectral) X-ray detectors have yet a limited total pixel number but an energy resolution of about 1 keV (Egan et al., 2015), allowing to distinguish even neighboring chemical elements. However, a high spectral energy resolution entails long exposure times since the acquired counts are distributed over multiple bins. Therefore, a state-of-the-art reliable denoising approach might help to improve the experimental throughput. In this study, to ensure strictly controlled conditions, I simulated the tomographic acquisition.

#### Data

I generated a volumetric phantom by combining several three-dimensional point clouds: two Swiss rolls, two moon crescents, and an s-curve. All point clouds were generated by the Scikit-Learn library (Pedregosa et al., 2011); to convert 2D point

clouds to 3D, the third axis was added by randomly sampling from the Uniform distribution. To convert the point clouds to a raster volume, I selected the size (in voxels) of each point. To resolve the ambiguous cases (when several materials appeared in the same voxel), I selected the priority order and always assigned the material with the highest order to fill the ambiguous voxel. The spatial size of the phantom was set to $512 \times 512 \times 512$. In a rough structure, all slices of the dataset are the same. However, the surface texture varied because of the random nature of the point clouds. A single slice is shown in Figure 3.2a (left).

I assigned the simulated objects with energy-dependent mass attenuation coefficient (MAC) of Europium ($_{63}\mathbf{Eu}$, k-edge = 48.5 keV), Gadolinium ($_{64}\mathbf{Gd}$, k-edge = 50.2 keV), Ytterbium ($_{70}\mathbf{Yb}$, k-edge = 61.3 keV), Lutetium ($_{71}\mathbf{Lu}$, k-edge = 63.3 keV), and Uranium ($_{92}\mathbf{U}$, k-edge = 115.6 keV). The background was assigned with MAC of air. This particular choice of materials was inspired by the study of the separability of k-edge nanoparticles presented in (Getzin et al., 2018). Two pairs of materials have neighbouring atomic numbers, hence very close k-edges, and are barely distinguishable in a noisy image; Uranium was added to have a k-edge in the noisiest part of the spectrum, to check the ability of the method to locate the k-edge in extreme noise conditions.

I used the MATLAB package `PhotonAttenuation` to generate the energy-dependent MAC of the selected materials (Tuszynski, 2006). A spectrum profile of a Boone/Fewell source with the tube potential 150 kV (no kV Ripple and filters) was generated using the MATLAB package `spektr` 3.0 (Punnoose et al., 2016). The obtained source spectrum was normalized and scaled to have a maximum value of $175 \times 10^3$ photons / mm$^2$ to imitate short exposure acquisition. MAC of selected materials and the source spectrum are shown in Figure 3.1.

I generated 135 energy bins between 15 and 150 keV with a 1 keV step. For each bin, I simulated 120 equally-spaced parallel-beam CT projections over 180 The spectral characteristics of the material and the source are shown in Figure 3.1. I used the conventional FBP algorithm to reconstruct tomographic data (implemented in (Jørgensen et al., 2021)). Examples of reconstructed slices for 40 keV (high flux) and 140 keV (low flux) are shown in Figure 3.2b (right). As expected, at 140 keV the reconstructed slice is uninterpretable.

### Training and Processing

The model $f_\theta$ was trained by optimizing

$$\mathbb{E}_{i,j} \| f_\theta(x_{i,j-1}, x_{i,j+1}) - x_{i,j} \|_1 \underset{\theta}{\rightarrow} \min, \tag{3.2}$$

where $x_{i,j}$ is a projection acquired at the transmission angle $i$ and in the energy bin $j$. I randomly split the whole set of projection angles into a training set and a validation set with a ratio of 80/20. I do not select a test set, since, in the experiments, I do want to overfit for the exact dataset and do not seek for generalization. Note that the energy level $j$, which is required to be predicted by a model, should not be fed into the model to avoid the trivial solution. Only adjacent $j-1, j+1$ levels should be used. This forms a gap of one energy level in the inputs.

During the inference, the model is fed with the adjacent energy bins without the gap used in training, to avoid blur in the spectral domain:

$$\tilde{x}_{i,j-0.5} = f_\theta(x_{i,j-1}, x_{i,j}). \tag{3.3}$$

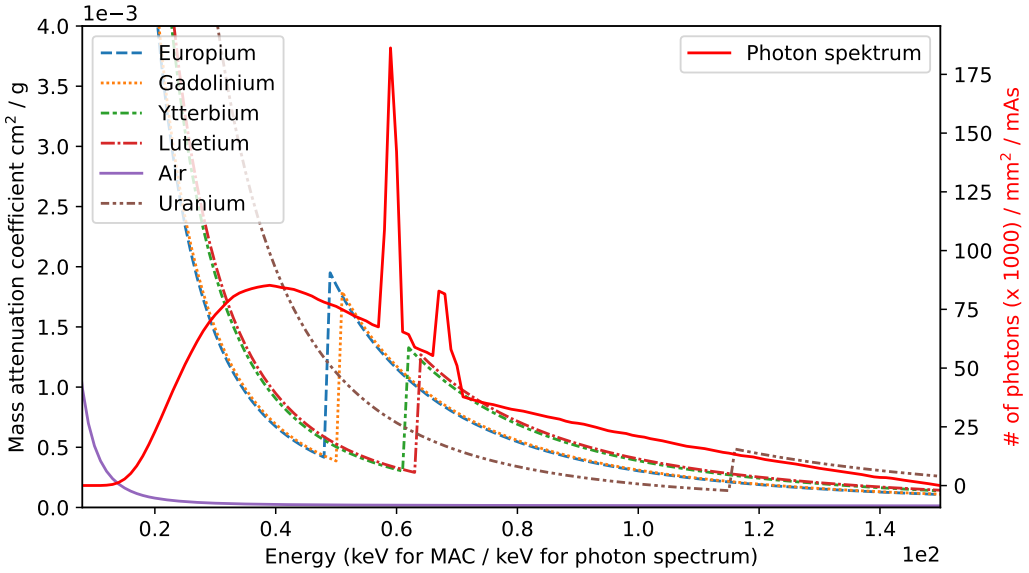



FIGURE 3.1: The materials and source characteristics used to simulate spectral CT. For materials, energy-dependent mass attenuation coefficients (MAC) are presented, and for the simulated Boone/Fewell source, I present the source profile. I selected two pairs of materials with close k-edges that are hard to resolve and one material with the k-edge in the low-flux zone of the source.

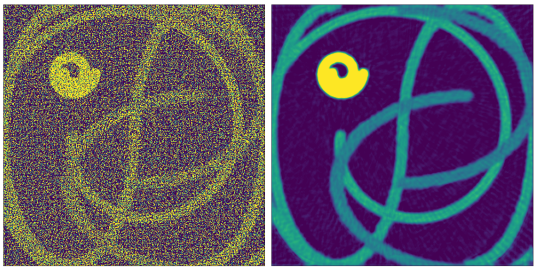
However, since the model predicts the energy level which is averaged between two input levels, it will inevitably predict an energy level between two adjacent ones used as input. It is important yet easy to compensate for this.

As before, the denoised tomographic datasets were reconstructed with the conventional FBP algorithm. Here, each energy bin was reconstructed separately resulting in 135 volumes. To obtain the spatial distribution of individual materials in the sample, I performed material decomposition as described in (Ametova et al., 2021b). The employed decomposition relies on the assumption that each voxel is a unit volume and each material occupies a volume fraction in this unit volume (the fraction can be 0). Under this assumption, a voxel-wise sum of all material maps is equal to 1 in each voxel.

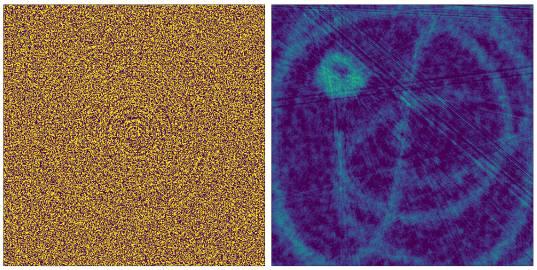
### Results

Figures 3.2a and 3.2b show two-dimensional slices for selected (individual) energy bins. N2N demonstrates the drastic quality improvement of the reconstruction. For 40 keV (high source flux, Figure 3.2a) the reconstructed slice appears to be almost noise-free; the slice shows sharply defined objects where all original structures become clearly visualized. Although no signal seems to be visible in the 140 keV slice (Figure 3.2b) prior to denoising, N2N is able to partially recover the structures in the slice.

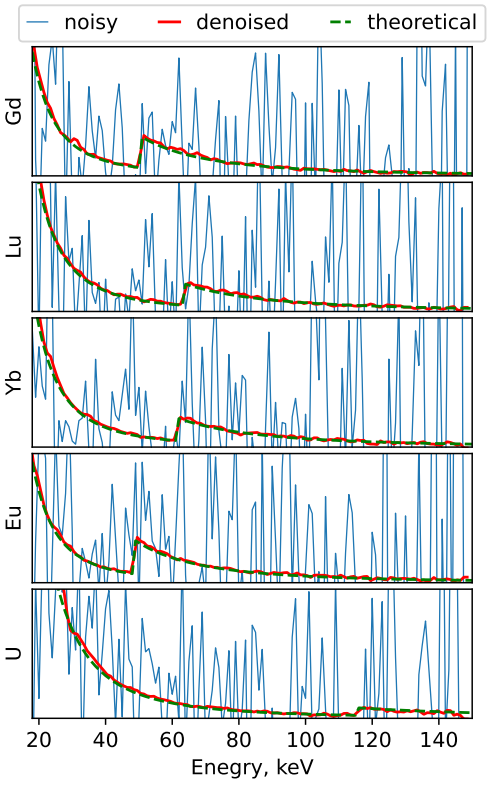
Single noisy energy spectra are used for one voxel per material component and denoised spectra are reconstructed for the voxels and plotted in Figure 3.2c along with the theoretical MAC. The voxel positions within the materials were chosen arbitrarily. Noise reduction results in sharp and accurately positioned k-edges, aiding further material decomposition. Even the slight uranium k-edge is visible in the denoised spectrum.



(A) Noisy and denoised slices at 40 keV (near peak source flux)



(B) Noisy and denoised slices at 140 keV (low source flux)




(C) Spectral profiles

FIGURE 3.2: Qualitative examination of the denoising of the simulated energy-resolved CT of a specially devised phantom. On the left, present a noisy (left) and denoised (right) transverse slice both near peak (top) and low (bottom) flux. On the right, I present a comparison of theoretical, noisy, and denoised spectra for different materials. For each material, I selected one representative pixel. Note how denoising is able to recover information even in extremely noisy cases both spatially (b) and spectrally (see the slight k-edge of the Uranium on the (c) plot).

| | | Air | Eu | Gd | Yb | Lu | U | mean |
|---|---|---|---|---|---|---|---|---|
| 2*AUPRC | noisy | 0.999 | 0.917 | 0.873 | 0.787 | 0.645 | 0.998 | 0.870 |
| | denoised | 0.999 | 0.998 | 0.998 | 0.996 | 0.995 | 0.999 | 0.998 |

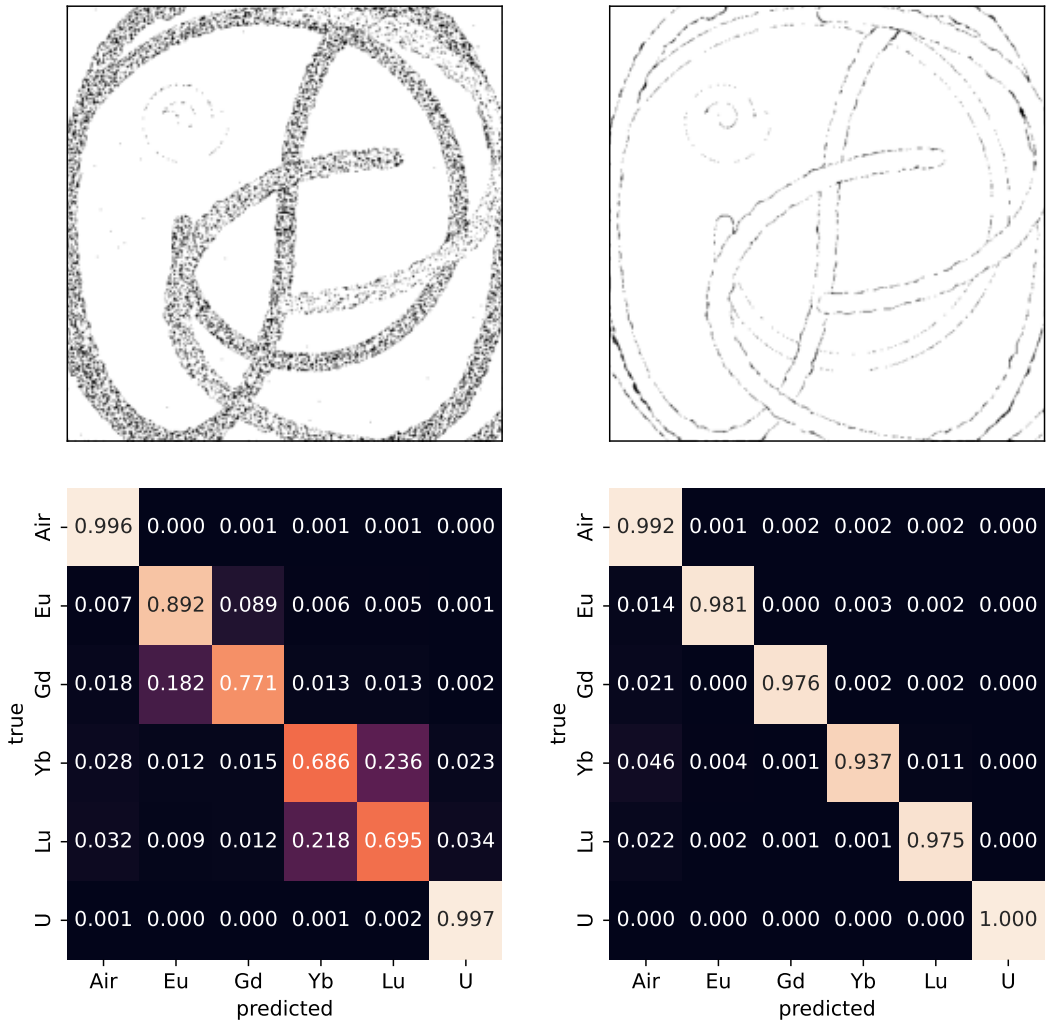
TABLE 3.1: Quantitative examination of the denoising of the simulated energy-resolved CT of a specially devised phantom. I numerically compare the material decomposition quality before and after denoising. The denoising provides a prominent quality boost for material decomposition.

To quantitatively evaluate denoising results, I perform the material decomposition. Since the sum of volume fractions corresponding to each material, obtained through material decomposition, is bound to 1 in each voxel, I can treat the estimated volume fractions as probabilities. Hence, the task of material decomposition can be considered a classification problem and the related quality assessment metrics can be applied to quantitatively assess the results. The comparison results are shown in the Figures 3.3a and 3.3b. In the top row I show the binarized material decomposition error (black corresponds to erroneous material prediction). The confusion matrices between the predicted and true materials for each pixel are presented in the bottom row (perfect classification results in the identity confusion matrix). A high level of noise in the simulated data causes misclassifications between close materials (e.g., Lutetium and Ytterbium). Also, as it is visible on the top row, these errors are distributed evenly throughout the sample. Hypothetically, this can be compensated by enforcing an assumption of material homogeneity. However, this assumption might cause severe errors close to material interfaces. The errors in the denoised volume are mostly concentrated around the borders (see the top row), and mainly correspond to misclassification for air due to slight blur (see the bottom row). But overall, the confusion matrix for the denoised dataset is considerably closer to the identity matrix.

I also present the Area Under Precision-Recall Curve (AUPRC), measured for each material Table 3.1. AUPRC for ideal classification is 1. AUPRC results additionally highlight improvement after denoising: N2N provides a boost of more than 10% of mean AUPRC for the downstream material decomposition. To assess quality loss caused by reconstruction itself (without any effect of denoising), I generated another set of projections with very high flux (all other parameters remained constant). Material decomposition for this volume shows a mean AUPRC of 0.999, with the lowest *precision* of 0.996 for the air. I can conclude that the reconstruction losses are negligible in this experiment.

### 3.4.2 Neutron imaging

As a second case study I discuss the applicability of N2N to energy-dispersive Bragg edge neutron tomography. Neutron imaging provides a complementary contrast to conventional X-ray imaging. Neutrons mainly interact with atomic nuclei, in this way a neutron beam passing through an object can capture information about the internal material structure. The energy spectrum of the neutron transmission of a polychromatic thermalized neutron beam passing a predominantly polycrystalline material contains sudden and sharp edges at wavelengths equal twice the interplanar distance between scattering planes in dependence of the crystalline properties




(A) Noisy (B) Denoised

FIGURE 3.3: Quantitative examination of the denoising of the simulated energy-resolved CT of a specially devised phantom. I study the quality of denoising through the lens of further material decomposition. On the top row, I present the binarized material decomposition error. Pixels that are black were assigned the wrong material. On the bottom row, I present the confusion matrix of the material decomposition for different materials, where ideal decomposition should yield the identity matrix. I note that materials with close k-edges are frequently confused before the denoising, and after the denoising, the confusion mainly comes from spatial smoothing.

of the sample material (Fundamentals, 1993). Energy dispersive images can efficiently be acquired by combining a pulsed neutron spallation source and a suitable time-sensitive detector by using the time-of-flight (ToF) method, which employs the energy-dependent neutron velocity for spectral information (the more energetic neutrons, by having higher velocities, reach the detector earlier than less energetic (slower) neutrons). Measuring the time of arrival of the neutrons at the detector and knowing the flight path length, their energies, and the corresponding wavelengths can be determined. For TOF methods, high energy resolution requires long flight distances and many time bins in the detector. Hence, only a few pulses per second can be measured, and acquired counts are shared between multiple bins (Santisteban et al., 2001). More details on this acquisition mode can be found elsewhere, for both the measurement setup (Kockelmann et al., 2007) and applications (Santisteban et al., 2002; Strobl et al., 2009). Neutron facilities are expensive and demand for neutron beamtime exceeds the supply capacity (Bentley, 2020). Therefore, there is a high interest in efficient image denoising techniques to reduce exposure time and subsequently increase experiment throughput.

#### Data

In this study, I employ the dataset (Jørgensen et al., 2019) acquired at the Imaging and Materials Science & Engineering (IMAT) beamline operating at the ISIS spallation neutron source (Rutherford Appleton Laboratory, U.K.) (Burca et al., 2013; Kockelmann et al., 2018). More details on acquisition parameters and preprocessing can be found elsewhere (Ametova et al., 2021b); here, I only briefly summarize details relevant to this study.

A sample contains 6 aluminium tubes: five filled with metallic powder (copper (Cu), aluminium (Al), zinc (Zn), iron (Fe), and nickel (Ni)), and one empty. The neutron detector has $512 \times 512$ pixels, 0.055 mm pixel size. A set of spectral projections were acquired at 120 equally-spaced angular positions over 180 degrees rotation with 15 min exposure. Additionally, 8 flat field images (4 before and 4 after the acquisition) were acquired with the same exposure.

A typical problem of spectral measurements is that noise statistics vary quite drastically across the spectrum. The beam spectrum at the IMAT beamline has a crude bell shape with a peak around 3 Å (Burca et al., 2013). Additionally, the time-sensitive detector suffers from dead time meaning counts loss, hence, additional signal distortions (Tremsin et al., 2012). To alleviate the count loss problem, the time (wavelength) domain is split in several independent measurement intervals (4 in this case) and a special correction technique is applied to the measured data (Tremsin et al., 2012). Each interval has an individual bin width; for this study the following bin width was used: $0.7184 \cdot 10^{-3}$ Å, $1.4368 \cdot 10^{-3}$ Å, $\cdot 10^{-3}$ Å and $2.8737 \cdot 10^{-3}$ Å. To benchmark N2N, I generated three additional datasets by rebinning the dataset in the original resolution (2840 energy bins split into 4 measurement intervals with (1141, 814, 424, 464) bins in each). The rebinning was performed individually in each interval by summing every (4, 2, 2, 1), (8, 4, 4, 2), and (16, 8, 8, 4) bins, resulting in datasets with 1366, 681 and 339 wavelength bins, respectively.

As a proxy to demonstrate the noisiness of the data, as a function of frequency, I plot the standard deviation of pixel values for one projection angle but different wavelengths along the spectrum in Figure 3.4. Vertical dashed lines separate independent intervals. Note, that standard deviation increases drastically with the increase of flux (the effect of counts loss becomes more apparent).

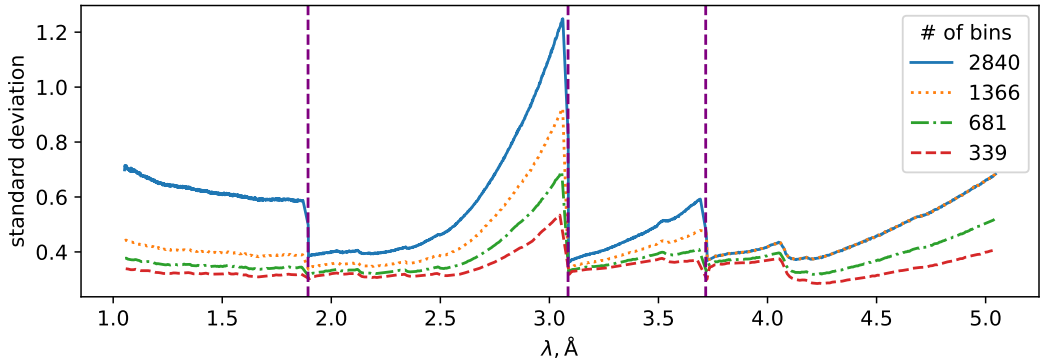



FIGURE 3.4: Limitations of time-sensitive detectors require splitting the whole wavelength domain into several measurement intervals (4 in this case, brackets depicted with dashed lines). Each interval has an individual wavelength bin width. In this plot, I show how standard deviation of the values captured for an individual pixel within the bin changes with the change in wavelength. This can be assumed to be a proxy measure of the signal noisines. Additionally, to benchmark the method, I generated three additional datasets by rebinning the spectral dimension of the original dataset.

### Training and Analysis Details

In this experiment, I compare the effect of noise reduction applied to the projections (N2N(P)) before reconstruction with that of applying it to the already reconstructed slices (N2N(S)). In both cases, I trained a model $f_\theta$ by performing essentially the same loss optimization procedure as in the previous case study

$$\mathbb{E}_{i,j}\|f_\theta(x_{i,j-1}, x_{i,j+1}) - x_{i,j}\|_1 \underset{\theta}{\rightarrow} \min, \tag{3.4}$$

where now $x_{i,j}$ can represent either the projection for an angle $i$ and an energy channel $j$, or a reconstructed slice number $i$ and an energy channel $j$. I used $i$ to randomly split the dataset into the training and validation subsets in the 80/20 ratio.

The combination of the N2N denoising approach and the conventional FBP reconstruction was compared with the advanced iterative reconstruction routine proposed by Ametova et al., 2021b. The latter relies on expert expectations on how the reconstructed image should look like. Which becomes increasingly complex with increasing complications of the sample under investigation. As in this case the reconstructed samples are expected to appear as solids, *i.e.* homogeneous regions, the authors assumed a piece-wise constant signal in the spatial domain. This prior knowledge is enforced through TV regularization (Rudin, Osher, and Fatemi, 1992; Sidky, Kao, and Pan, 2006). The signal in the spectral domain is expected to be piece-wise smooth based on theoretical predictions for the materials employed in this study (Boin, 2012). In this case, regularization is achieved through Total Generalized Variation (TGV) prior (Bredies, Kunisch, and Pock, 2010). Hence, I refer to the iterative reconstruction method as TV-TGV. As before, the reconstruction was implemented in CIL (Papoutsellis et al., 2021). Code to reproduce results is available from (Ametova et al., 2021a).

**Results Discussion**

I will begin with a visual comparison of different denoising approaches in the spectral domain (Figure 3.5). I perform the comparison for the 339 channels image as the same binning was used for the case study in Ametova et al., 2021b. The theoretical predictions provide the ground truth for the comparison. As in the previous case study, without denoising the conventional FBP reconstruction results are uninterpretable. The N2N performance is comparable to a TV-TGV reconstruction. TV-TGV provides smoother spectra at a cost of spectral and spatial resolution loss. In contrast, N2N results appear sharper spatially but noisier spectrally for low-attenuative materials. Hence, I conclude that there is a certain threshold noise level N2N can handle efficiently.

Figure 3.6a shows a comparison of the slices reconstructed from the white beam data (sum of all energies) and from data for a selected single energy channel for TV-TGV, N2N(S), and N2N(P), through direct comparisons of reconstructed slices in the transverse plane. While for Fe and Ni, both N2N(P) and N2N(S) perform comparably, for Cu and Al their performance differs. The attenuation of Al is drastically lower than other materials, which could lead to inconsistent predictions of the model for the projections when another material occludes the Al cylinder. This problem is not relevant for N2N(S). The Cu powder has a larger mean particle size than other powders (the mean particle size is comparable to the voxel size), hence, stronger spatial structures are visible in the cross-section. The structure changes randomly along the sample height. Therefore the N2N(S) model has less information about the structure and might fail to recover it correctly.

As a reference revealing structures, I use an FBP slice averaged across all energy levels, sacrificing spectral information for spatial. I also report the structural similarity index (SSIM) between the single-energy slices and the reference slice (Wang et al., 2004). Both N2N approaches provide a sharper, more detailed image than TV-TGV. Interestingly, while N2N(S) provides a visually better, sharper image, this image has lower SSIM, compared to the N2N(P). I hypothesize, that this is caused by the unintentional reduction of the streak artifacts (highlighted in the top left callout in the N2N(P) slice). Streak artifacts are very common in tomographic imaging and are caused by insufficient angular sampling (Kak and Slaney, 2001).

I next explore denoising quality in the spatial (Figure 3.6c) and the spectral domain (Figure 3.6b) given the increase of noise levels in the input data. I control noise levels by changing binning: the smaller is the binning step–the lower is SNR. I use the white beam slice reconstructed with FBP and the theoretical predictions for SSIM calculations in the spatial and spectral domains, respectively. While iterative reconstruction provides the best results for the spectral domain, it provides the worst result for the spatial domain. Excellent TV-TGV performance heavily capitalizes on the fact that the cylinders are homogeneous inside. In terms of SSIM, N2N(P) outperforms N2N(S) because N2N(S) additionally minimizes streak artifacts due to the angular undersampling, hence, the discrepancy between the reference image and the denoised one grows.

Another important observation is that N2N can be computed for the higher number of channels. The training time of the model stays almost the same, around 20 hours on average for the full volume, calculated on a $4 \times A5000$ machine. After the training, the model is capable of inferencing one projection/slice at the rate of 20-30 energy channels per second. While TV-TGV reconstruction for one slice with 339 channels takes several hours to complete and reconstruction time increases with the increase in the number of channels or the number of slices.

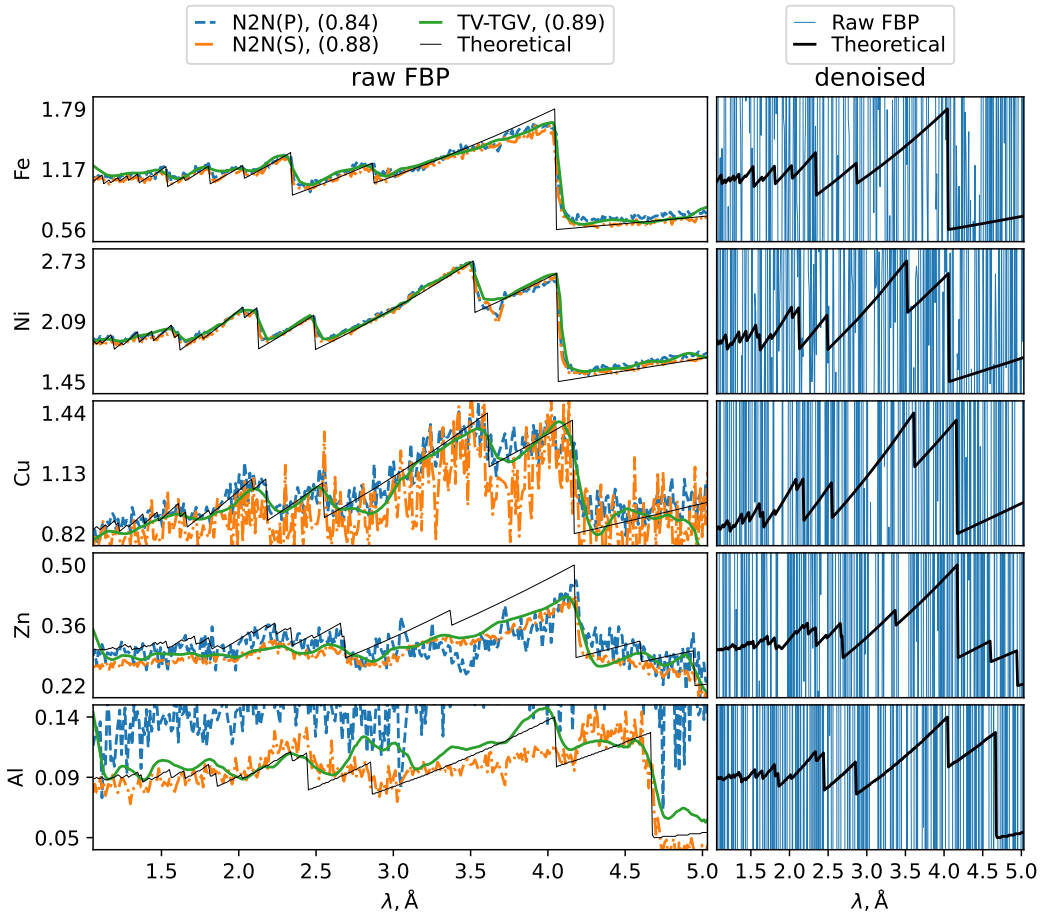
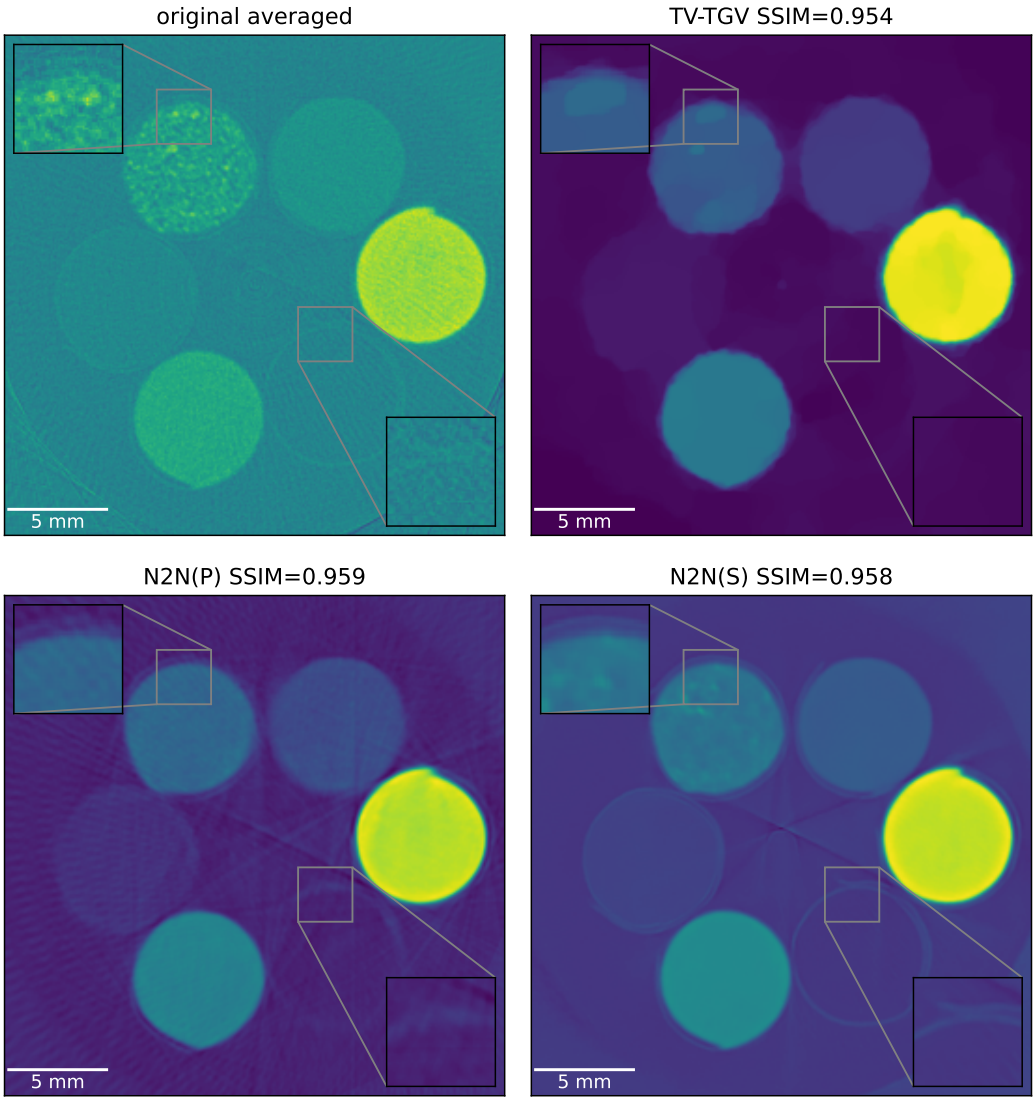
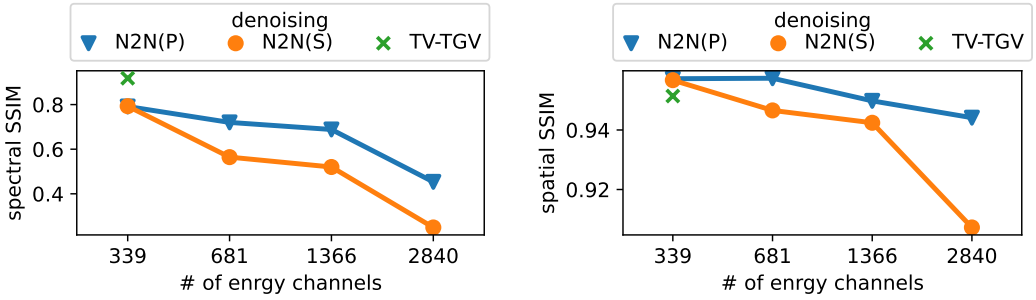



FIGURE 3.5: Qualitative comparison of denoising techniques in the spectral domain for the neutron imaging dataset. The $\lambda$ designates the wavelength, and is measured in angstroms. For each material, I selected one representative voxel, and present the theoretical and empirical spectra. Left: results of TV-TGV, N2N done on slices, and N2N done on projections are presented; right: the spectra before denoising are presented. All results are presented for the datatset with 339 energy bins. I can note, that N2N provides sharper edges, but noisier predictions.




(A) Transverse slices showing white beam (sum of all energies) and single channel at the energy level of 1.63 Åreconstruction for TV-TGV, N2N(S), and N2N(P).




(B) SSIM between predicted and experimental spectra.

(C) Averaged SSIM between denoised slices and white beam slice reconstructed with FBP.

FIGURE 3.6: Qualitative (top) and quantitative (bottom) comparisons of the denoising methods for the neuron imaging are presented. For the quantitative comparison, I plot the dependency between the structural similarity index (separately in spectral and spatial domains) and the number of energy channels used. Since the change in the number of channels was done through the binning, the lower amount of channels corresponds to the lower amount of noise in the initial image. I note, that spatially Noise2Noise provides superior denoising.

### 3.4.3 *In Vivo* Cine-Radiography

The third case study considers a N2N application to cine-radiography. Digital real-time radioscopy, cine-radiography alias fluoroscopy are different realizations of time-resolved X-ray imaging techniques relying on radiographic projection imaging to study morphological evolution during technological or biological processes. In particular, for *in vivo* or other dose-sensitive applications, the applicable dose and the detection efficiency of the imaging system limits acquisition times, constraining the total observation time or achievable SNR.

For this case study, I employed propagations-based phase contrast imaging (PB-PCI), which is particularly well suited for X-ray imaging of very weakly absorbing soft tissue in biological specimens in the sub-micron up to a few µm resolution range (Fitzgerald, 2000). The X-ray wavefield experiences a locally varying phase shift when traversing the specimen, which turns into measurable intensity contrast as a result of free-space wavefield propagation. The object information can be reconstructed from the detected image interference pattern by algorithmic treatments (so-called *phase retrieval* or PR for short (Lohse et al., 2020)). Here, I applied a convolution with a dedicated low-pass filter in the spatial domain. This so-called Paganin filter (Paganin et al., 2002) heavily affects the noise distribution. On one hand, it significantly reduces high-frequency noise, hence, increases the Peak Signal-to-Noise Ratio (PSNR). On the other hand, low-frequency noise becomes more prominent causing so-called "cloudy" artifacts (Paganin et al., 2004). For a single image, the effect of low-frequency noise might be less disturbing. However, in a time-resolved cine-radiographic sequence, this effect leads to a highly disturbing flickering, since the position of these "clouds" changes randomly from frame to frame, which affects the interpretability of the images by experts.

#### Data

In this case study, I used a batch of *in vivo* cine-radiographic data from a behavioral study visualizing the morphodynamics of parasitoid chalcid wasps emerging from their host eggs (Spiecker et al., 2023a). The full dataset contained 138 videos, imaged with 15 fps (0.066 s exposure time per frame) with lengths between 81 and 7142 frames per image series. The total number of frames is 263,875.

I identified a sequence of 100 frames, where the wasp was completely still. From this, I calculated an average frame and used it as a low-noise reference image. This averaged image was used for qualitative results calculations. The average PSNR value before phase retrieval is 25.2 with a standard deviation of 0.02. After the Paganin phase retrieval, the PSNR increases to 35.9 with a standard deviation of 1.2.

#### Training and Analysis Details

Because of the high dynamics in the sample's motions, I cannot use more than one frame as model input at one pass. I train the model $f_\theta$ by optimizing the loss

$$\mathbb{E}_{i,j}\|f_\theta(x_{i,j-1}) - x_{i,j}\|_1 \underset{\theta}{\rightarrow} \min, \tag{3.5}$$

where $x_{i,j}$ stands for the frame number $j$ from the image sequence number $i$. I have randomly divided all frames into training and validation sets in the 80/20 ratio according to the index $i$. In addition, I noticed that in some cases the temporal resolution was not high enough to smoothly capture fast movements because the structure positions changed significantly between adjacent frames. I introduced additional

| | measured before PR | | measured after PR | |
|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM |
| no denoising | $25.2 \pm 4 \times 10^{-3}$ | $0.41 \pm 1 \times 10^{-4}$ | $36.0 \pm 0.2$ | $0.97 \pm 2 \times 10^{-3}$ |
| denoising before PR | $33.1 \pm 13 \times 10^{-3}$ | $0.49 \pm 3 \times 10^{-4}$ | $37.3 \pm 0.3$ | $0.98 \pm 1 \times 10^{-3}$ |
| denoising after PR | - | - | $36.0 \pm 0.3$ | $0.97 \pm 1 \times 10^{-3}$ |

TABLE 3.2: Quantitative comparison of the denoising done before and after phase retrieval for the chalcid wasp cine-radiography. I averaged 100 motion-free frames to use as the reference (noise-free) image for these calculations. I report mean values and 95% confidence intervals. The denoising before phase retrieval provides a slight improvement in measures both before and after phase retrieval.

filtering to alleviate potential blur caused by the large morphodynamical changes between neighboring frames. During the training, I discard the image pairs whose SSIM was below a manually optimized threshold.

**Results**

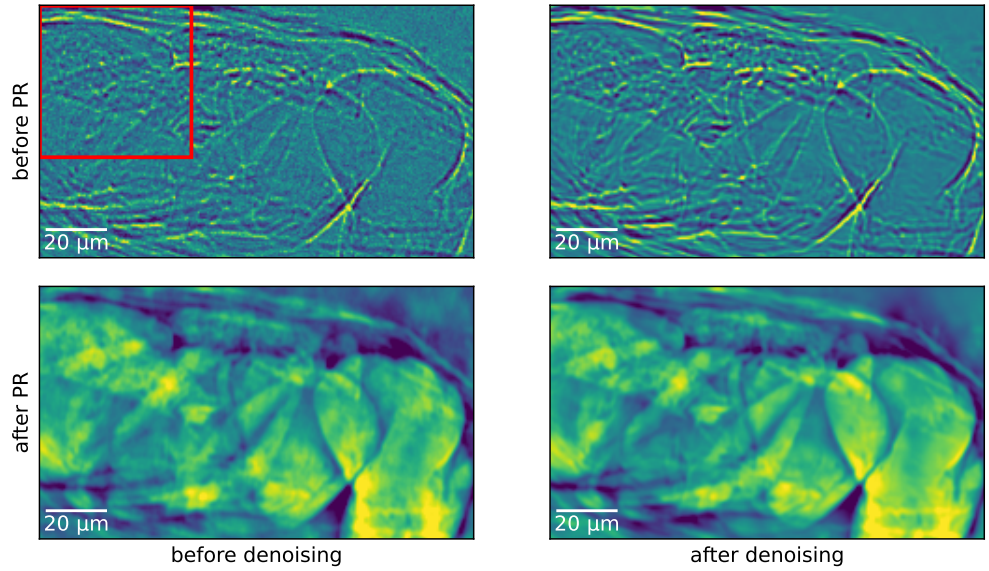
I applied the N2N denoising once before and once after phase retrieval. Table 3.2 summarizes PSNR and SSIM for both cases (the average of 100 frames without motion was used as a reference for metrics calculation). Applying denoising before the phase retrieval results in significant improvement in PSNR and SSIM. The benefits are maintained even after phase retrieval.

To qualitatively assess the benefits of denoising done before the phase retrieval, I show exemplary frames in Figure 3.7. Note that after the denoising and before phase retrieval the complex structures of the insect leg and interference fringes become more visible (Figure 3.7b). I also visually compare how the noise changes between consequent frames without (Figure 3.7c) and with (Figure 3.7d) denoising. I note that the noise not only becomes less sharp without blurring the sample (Figure 3.7a) but also produces less sudden changes in consequent frames. This makes it easier to evaluate the morphodynamics or, reversely, would allow reducing the dose even further. While denoising made the images smoother, there is no drastic blur, and even relatively small details (*e.g.*, legs or antennae) are preserved.

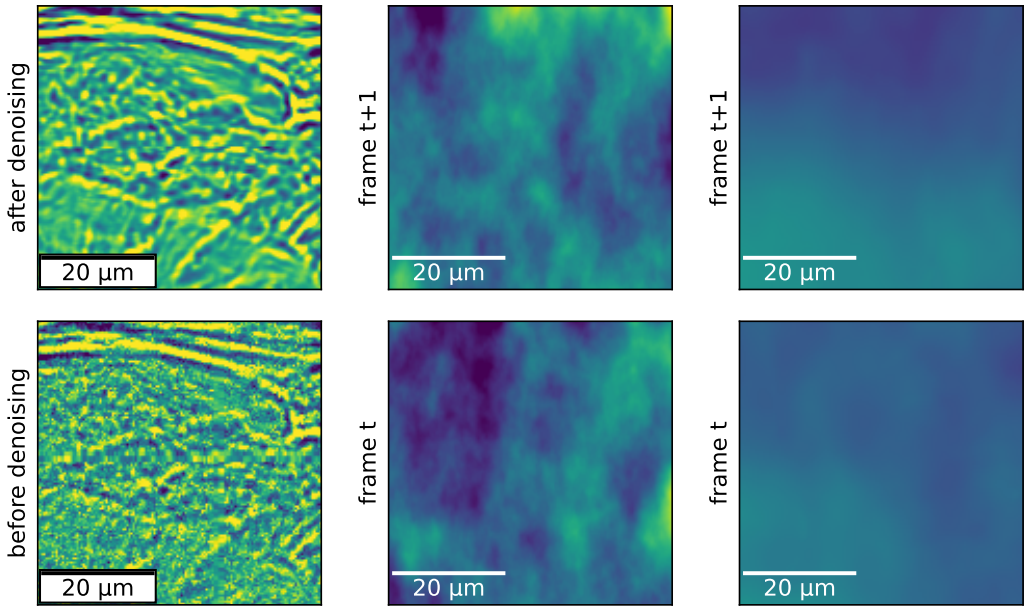
## 3.5 Discussion

In this chapter, I proposed and tested a way to relax the data constraints of the N2N method. I traded the degree of similarity required from signals of individual image pairs for the number with structurally close image pairs. That is, instead of taking one pair with an identical signal, application to multi-channel imaging allows us to benefit from drawing tens or hundreds of pairs with close but not identical signals. Through the experiments with different imaging modalities, I demonstrated the method's capability to significantly enhance image quality without oversmoothing along the energy or time domain. I also highlighted the vulnerability of the method to significantly dissimilar training image pairs (Section 3.4.3). I suggested an approach to overcoming this issue by discarding images based on some image similarity metrics.

In general, the method does not compensate for systematic artifacts present in all channels (energy bins or time frames), hence relevant corrections (such as centre-of-rotation compensation in CT) remain essential. On the other hand, I noticed that




(A) Overall view of the wasp.




(B) Leg of the wasp before PR. (C) Consequent frames before denoising. (D) Consequent frames after denoising.

FIGURE 3.7: Qualitative examination of the denoising performed for the chalcid wasp cine-radiography. I show the same cropped slice before and after the denoising. I also compare results before and after the phase retrieval. Denoising is done only before phase retrieval. In (b), I demonstrate an enlarged view of the wasp's leg before PR (a callout from the red rectangle in (a)). In (c) and (d), I show consequent frames of the noise without the sample, before and after denoising. All four frames are plotted with the same value range. This demonstrates that not only the noise becomes less prominent, but also the evolution of the cloudy noise becomes less drastic after the denoising.

N2N applied to reconstructed slices significantly reduced the appearance of the ring artifacts (Figure 3.2b) and the undersampling artifacts (Figure 3.6a). The magnitude and the limits of this secondary effect are in the scope of future research.

I yet observed superficial "cloudy" artifacts present in homogeneous image regions (such as the background) after denoising. These artifacts do not have any significant effect on CT data as they have lower contrast than actual image features. However, in cine-radiographic time series, they affect the overall image perception as their location changes randomly from frame to frame introducing strongly disturbing flickering without denoising. The reduced flickering after denoising improves image interpretability as it has very low contrast compared to image features and becomes more of a cosmetic effect as the human eye is still sensitive to it. A way around this is to generate a few images for each frame by adding some Gaussian noise, i.e. to increase the noise level, and take a median value of the resulting denoised images. However, increasing the number of used denoised images or the variance of the noise leads to blurring. I set these parameters, guided by the expert judgment on the resulting image.

N2N assumes that the image pairs have equal signal values and independent noise drawn from the same distribution. Strictly speaking, both assumptions might be violated in spectral and time-resolved imaging where values in each individual channel are either energy or time-dependent, and noise distribution might be partially correlated (see data discussion in 3.4.2 for details). The application of the method to such data is based on the assumption that the variability of noise between the twin images is larger than the variability of the signal.

In spectral CT, N2N can be applied to both projection images and to tomographic slices after reconstruction. Any corrections in the projection domain are challenging as they might cause or exaggerate existing inconsistency between projections (a consistent sinogram has strong restrictions expressed as Helgason–Ludwig consistency condition (Helgason, 1965)). However, the empirical studies did not show any noticeable artifacts due to this inconsistency.

# Chapter 4

# Optimizing the Markup Preparation Procedure

This chapter explores the critical challenge of data annotation in the field of neural networks for computer vision, specifically within the context of Computed Tomography (CT) segmentation. Despite technological advancements easing the burdens of data collection and sharing, the intensive requirement for high-quality, expert-annotated datasets remains a significant hurdle, particularly in specialized fields such as medical imaging. The involvement of subject matter experts, such as medical practitioners, in the data labeling process, introduces a bottleneck due to the expertise and time required, potentially compromising the primary duties of these professionals and impacting the quality of dataset annotations. In this chapter, I've tried to answer the question what is the optimal dataset, define it's characteristics and find how change in those characteristics drives the model quality.

## 4.1 Introduction

Neural networks for computer vision have made significant strides, achieving results that rival those of human experts in certain tasks (Avetisian et al., 2020). This remarkable level of performance, however, is predicated on the availability of vast quantities of training data. As a consequence, there is an ongoing and intense pressure to collect and label ever-larger datasets to sustain and further enhance the performance of these models.

With the advent of technological advancements, the burden of data collection and sharing is steadily diminishing. Automation continues to expedite the process of data collection, making it feasible to gather massive datasets in relatively short periods. Simultaneously, the continuous decline in storage costs, coupled with the accelerated data transfer speeds offered by modern networks, has significantly eased the challenges associated with data sharing and storage. Hence, the constraints related to data collection and dissemination are becoming less obstructive in the landscape of neural networks and computer vision.

However, a new challenge has arisen in the form of data annotation, especially as the resolution of data expands and tasks become more complex. The task of labeling high-resolution, complex datasets often demands a high degree of expertise and is a time-consuming process. The labeling process becomes even more challenging when it necessitates the involvement of subject matter experts. For instance, in medical imaging, the labeling process often requires input from medical practitioners who possess the necessary knowledge to accurately annotate the images.

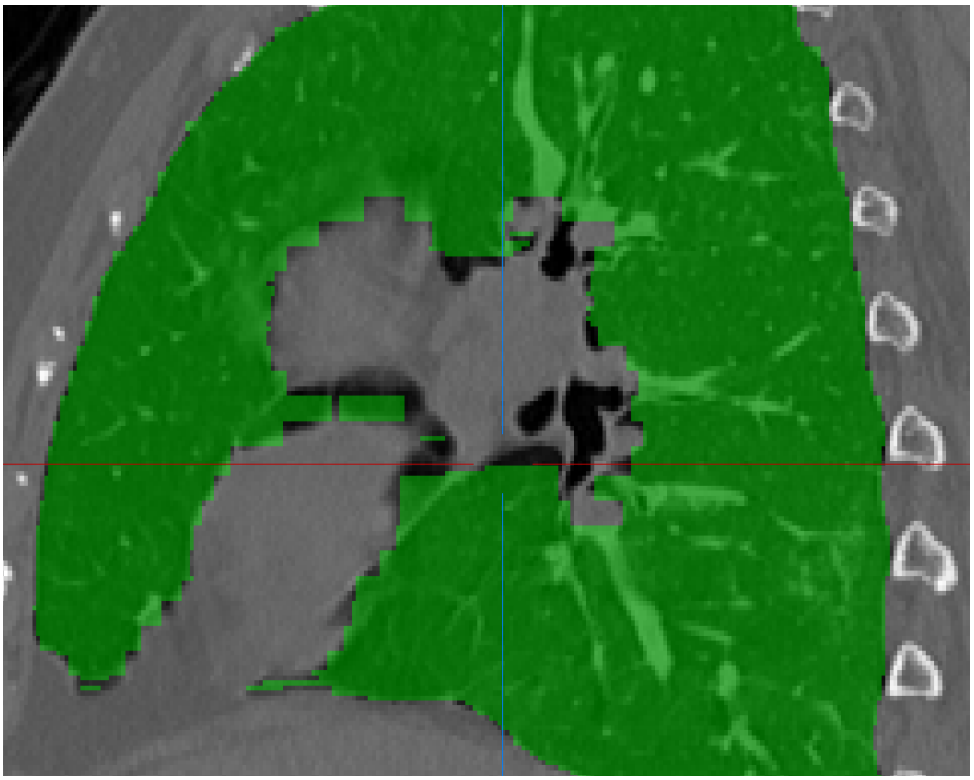


FIGURE 4.1: Exemplar image of the bad interpolation of the human lung segmentation, as found in Ma et al., 2021.

The need for experts' involvement in the labeling process introduces another level of complexity. Medical practitioners, already burdened with their primary responsibilities, are now expected to devote considerable amounts of their time to act as labeling experts. This requirement of experts spending many hours annotating datasets comes at the expense of their primary duties and is a significant hurdle in the development and deployment of advanced neural networks in fields such as healthcare. This sometimes leaves us with the markup of the undesired quality, e.g., Figure 4.1.

This conundrum underscores the need for innovative solutions in neural network training that can minimize the reliance on large, expert-annotated datasets. It also highlights the importance of developing models that can generalize well from smaller, meticulously annotated datasets, thereby alleviating the pressure on experts and making the process of deploying advanced neural networks more efficient and sustainable.

The quest to minimize the volume of labeled data required for training machine learning models has given rise to several lines of research. Approaches such as transfer learning (Zhuang et al., 2021), self-supervised learning (Jing and Tian, 2021), and active learning (Ren et al., 2022) have been developed with this very goal in mind. Of these methods, only active learning seeks to modify the labeling process itself, while the others predominantly operate under the assumption that a given dataset is available.

In the realm of Computed Tomography (CT), the pool of available datasets is experiencing rapid growth, propelled by advances in instrumentation and the rise of efficient digital pixel array detectors (Kiryati and Landau, 2021). However, the inherent diversity of biological or technical specimens, coupled with stringent data

requirements specific to the medical field (such as ensuring anonymity and accurate representation of diseases), often necessitates the collection of task-specific datasets.

Given this rising demand for unique, task-specific datasets, it is crucial for us, as a research community, to consider how we might go about acquiring the best possible dataset with the least amount of effort. To understand what constitutes the 'best' dataset, it's important to note that the process of collecting a dataset is not the final destination but rather a means to an end - the production of a high-performing model. Therefore, the value of a collected dataset can be gauged by the performance of the model that is trained on it.

While this work primarily focuses on the specific task of medical CT segmentation, the conclusions I draw are applicable to other CT segmentation tasks as well. This is because the approach does not impose any assumptions that are uniquely specific to medical imaging data. I believe that the insights gleaned from my research will have broader relevance and applicability across a range of CT segmentation tasks.

In my view, the three fundamental characteristics of an optimal dataset for training purposes are its quality, diversity, and completeness. The attribute of *quality* can be further bifurcated into two separate facets, those being the quality of the data itself and the quality of the labels. While I discuss datasets assembled specifically for model training, the quality of the data is dictated by the intended application of the model. As such, I will be focusing less on data quality in this discussion. Label quality, on the other hand, is primarily influenced by the level of precision of the annotation. More specifically, it refers to the accuracy with which the segmentation labels align with the actual anatomy present in the images (Lösel et al., 2020). Thus, going forward, any reference to *quality* will imply *label quality*.

*Diversity* in this context signifies the ability of the dataset to encapsulate a wide range of sample variations that are regulated by known parameters, such as a patient's age, sex, and medical history. On the other hand, *completeness* pertains to the capacity of the dataset to represent the naturally occurring variances in human morphology. For instance, even identical twins may present subtle differences in their morphological features.

Taking the perspective of the data manifold into account, *diversity* is characterized by the ability to provide sparse coverage of the entire manifold with representative prototypical examples. Contrarily, *completeness* is defined by the ability to densely populate the manifold and faithfully represent its distribution nuances. In essence, while diversity ensures a broad spectrum of examples, completeness ensures a rich, detailed portrayal of each instance within that spectrum.

In this work, my primary goal is to determine the relative impact of these three dataset virtues—quality, diversity, and completeness—on model performance. Theoretically, if a dataset perfectly embodies all three of these characteristics to the maximum possible degree, there would be no need for a model. Every conceivable instance would already be accounted for within the dataset, thereby transforming the prediction function into a simple lookup table.

However, in real-world scenarios, datasets rarely reach this ideal. Instead, models are required to interpolate the full data manifold from sparse points provided in the training dataset. Certainly, enhancement in any of the three virtues leads to the creation of a superior model. Nevertheless, with time as a limiting factor, domain experts must judiciously balance these three virtues. This process is often guided by intuition, and sometimes these decisions are even made unconsciously.

In many circumstances, the balance is more art than science—more intuition than procedure. This can lead to suboptimal results, as intuition is not always correct, and

implicit decisions can overlook important considerations. Therefore, a more systematic understanding of the interplay between these three virtues—quality, diversity, and completeness—in different contexts and for various tasks would be valuable. This could help experts make more informed decisions about how to best allocate their limited time and resources when assembling a training dataset. This work is a step towards a more quantified, more systematic understanding of the interrelationship between these virtues in the context of training data collection.

In contrast to the concurrent work by Kim et al., 2022, I focus on the segmentation task specifically, and consider only the fully supervised training, as opposed to the weak supervision as a way to vary label granularity. I also propose a labeling procedure that optimizes the effort.

## 4.2 Method

### 4.2.1 Datasets Preparation & Model Training

In this study I choose the brain tumor, heart, and liver tasks from the Medical Decathlon segmentation datasets (Antonelli et al., 2022). For the sake of simplicity, I joined all available classes, to represent binary segmentation, however, my private experience shows that the results hold for a multiclass segmentation. For each dataset, I took the openly available markup and split substracted 20% of the available data as the `test` set. The `test` is selected once for a dataset and never altered. I call the other 80% as `train+val` set, as it will be split again later on.

In medical datasets, it is typical to have a relatively small amount of volumes collected from a representative variety of patients (Luca et al., 2022). Hence, I assume that the portion of random volumes used for training could be a proxy to the *diversity*, and I specify it as a number $\in (0, 1]$ representing this portion. Although this subsampling also affects *completeness*, as I show in Section 4.3.3 and especially in Figure 4.6, the model responds differently to diversity and completeness. I conclude that this is a plausible and sufficient proxy for the purposes of qualitative comparisons presented in this chapter. I follow Zettler and Mastmeyer, 2021, and always train a 2D model on slices, instead of a 3D one on volumes. Based on assumption that adjacent slices represent small variations of roughly the same morphology, I use the portion of the slices used for training as a proxy for the *completeness*, which is reported as number $\in (0, 1]$ as well. Finally, as a proxy for the *quality* of the dataset, I take a subset of equidistant slices and interpolate the labels between them using the nearest neighbor approach. Varying the distance between slices, and therefore the interpolation errors, allows us to manipulate the label *quality*, which I report as a percent $\in [0, 100]$ representing the IoU between the label after interpolation and the original label.

To measure the model performance for some virtue value, I:

1. modify the `train+val` part of the dataset to model some virtue:
   - sample a portion of volumes to model *diversity*;
   - sample a portion of slices containing a mask to model *completeness*;
   - sample an equidistant set of slices and interpolate markup between them to model *quality*.
2. split the resulting data into `train` and `val`, at a ratio of 80 to 20.

3. upsample the `train` in such a way, that the amount of labeled slices is always equal to 80% of labeled slices in the original `train+val` set;

4. fit the model on `train`, select the best snapshot on `val`, and measure the model quality on `test`.

I hypothesized that, while tuning the model and optimizer hyperparameters can change the model performance, it will not change the relative importance of different dataset virtues for the model performance. Therefore, I always train the same model (UNet (Ronneberger, Fischer, and Brox, 2015) with ResNet-18 (He et al., 2016a) as the backbone), with the same optimizer (Adam (Kingma and Ba, 2015) with $3e-4$ learning rate), for the same amount of epochs (100 epochs and 10 epochs long cooldown of the early stopping). For each measurement, the median of 5 runs on random `train+val` splits is reported.

### 4.2.2 Results Interpretation

The target of optimization of the labeling procedure is to obtain the model with the best performance given a certain available effort budget for labeling. For example, an expert can roughly segment 10 volumes, or spend the same time, precisely segmenting 3 volumes. For experiments, I devise custom proxies of the effort measure. I leave the empirical measurement of the effort (e.g., as elapsed time) for future research, however, I consulted with experts involved in the segmentation process, and they concur with my estimation.

To find the optimal strategy I consider a plot, where the horizontal axis describes labeling efforts spent, and the vertical axis represents the model performance (to make the plots clearer I normalize the model performance to the quality of the model trained on the unaltered data). For the same amount of effort spent pursuing different virtues, I will have different model qualities. The optimal strategy of labeling is represented by a convex polyline that passes through the points on the plot in such a way that no points lie above it. Following this trajectory provides the best possible dataset at any given moment.

To understand the optimal strategy, I consider another plot. On the horizontal axis, I plot the model performance, and on the vertical axes–the value of the compared virtues. For each point on the optimal trajectory, I add one point per virtue in comparison. Therefore, each vertically aligned set of points represents the virtues required to achieve a specific model performance. This way, moving along the horizontal axis, I can see which virtue should be pursued earlier on, to stay on the optimal trajectory.

## 4.3 Results

### 4.3.1 What is More Important, *Quality* or *Diversity*?

To compare *quality* and *diversity*, I define effort as the portion of the volumes used (as a measure of *diversity*) multiplied by *quality*. E.g., 0.1 of the volumes segmented with 80% IoU will result in 8% effort.

The sampled plot with the optimal trajectory is shown in Figure 4.2. From this plot, I observe that the optimal trajectory connects the high-quality points, while low-quality points always fall far below the line.

I show in Figure 4.3 how *quality* and *diversity* drive the optimal trajectory. From this plot I conclude, that *quality* is more important early on, even though I never

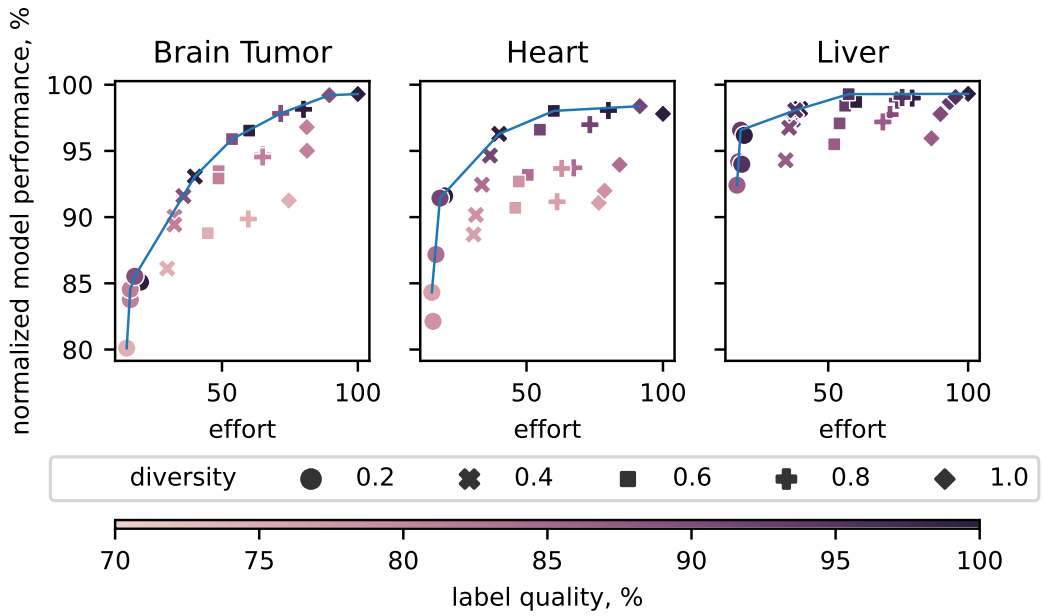



FIGURE 4.2: Optimal trajectory w.r.t. *quality* and *diversity* of a dataset. The variation of the *quality* is represented by color, of *diversity* by the marker shape. Based on the trajectory I can observe that the optimal trajectory connects the high-*quality* points, while low-quality points always fall far below the line.

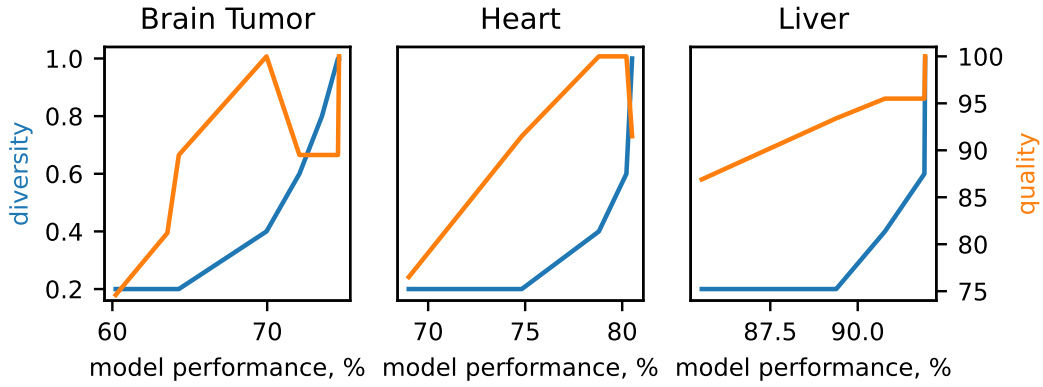



FIGURE 4.3: Importance of the *diversity* and *quality*. The orange line represents label *quality*, and blue–*diversity*. From this plot I can conclude that the *quality* is more important than *diversity* early on during the labeling process.

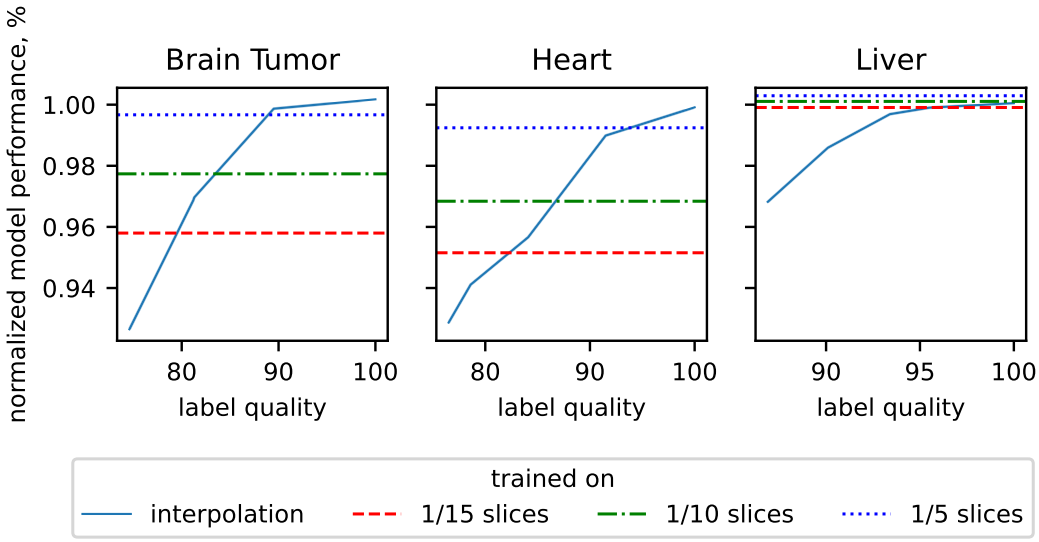



FIGURE 4.4: Plateauing of the model performance with the increase of the label *quality*. Additionally, the performance of the model trained on a small portion of 1.0 quality slices is represented with horizontal lines. From this plot I can conclude that the model's performance surpasses the sparse labeling of 10% of slices only with 90% label quality, and starts to plateau further on.

use IoU worse than 75% (which could be admissible quality for small area labels). However, as *quality* reaches around 90%, increasing *diversity* becomes as important or even more important than increasing *quality*.

### 4.3.2 How Much Labeling *Quality* is Enough?

Increasing labeling quality up to 100% is challenging if not impossible. But where is a meaningful threshold of the labeling quality, after which the model performance stagnates? To investigate it, I plot the model performance against the labeling quality (see Figure 4.4). I also plot the performance of the model trained on sparsely segmented slices (each 5th, 10th, and 15th). I conclude, that, first, if one can not achieve good interpolation quality, it may be even harmful to interpolate, and, second, in accordance with the previous section–one should aim for 90% quality of the labeling before aiming for either *completeness* or *diversity*.

### 4.3.3 What is More Important, *Diversity* or *Completeness*?

To compare *diversity* and *completeness*, I define the effort as a total percentage of slices segmented. E.g., if I sample 6 volumes from 10 available (*diversity* = 0.6) and segmented each 10th slice (*completeness* = 0.1), then the effort is 0.06.

The optimal trajectory plot is shown in Figure 4.5. The brightest demonstration of the importance of *diversity* is in the right bottom parts: the 0.2 of *diversity* with 1 of *completeness* is much worse than vice versa.

In Figure 4.6 I demonstrate the importance of *diversity* and *completeness* for the optimal trajectory. Not only *diversity* is more important early on, but *completeness* contributes less to the model performance (note the steeper growth of the *completeness* with the performance growth). Therefore, finding more diverse examples and

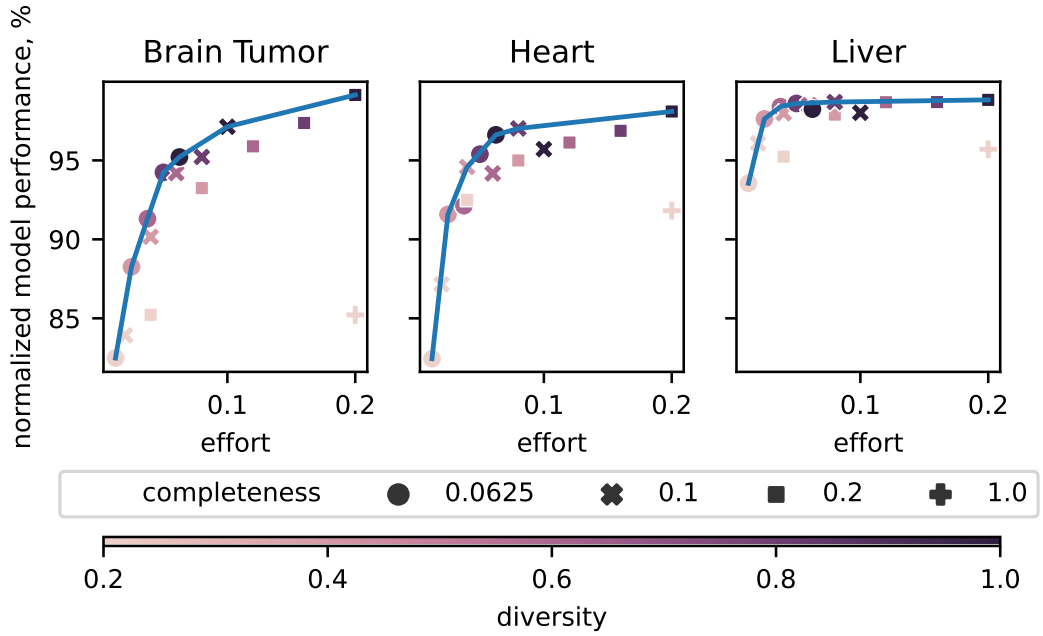



FIGURE 4.5: Optimal trajectory w.r.t. *completeness* and *diversity*. Variation of *diversity* is represented by color, of *completeness* by the marker shape.The brightest demonstration of the importance of *diversity* is in the right bottom parts: the 0.2 of *diversity* with 1 of *completeness* is much worse than vice versa.

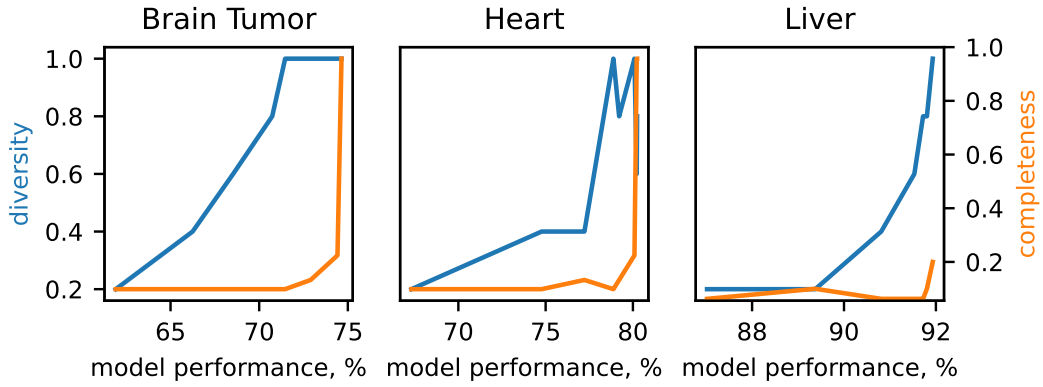



FIGURE 4.6: Importance of the *diversity* and *completeness*. The orange line represents label *completeness*, and blue–*diversity*. By the steep angle of the *completeness* during the late model performance increase stages, I note that the *diversity* is not only more important early on, but brings more impact overall, than *completeness*.

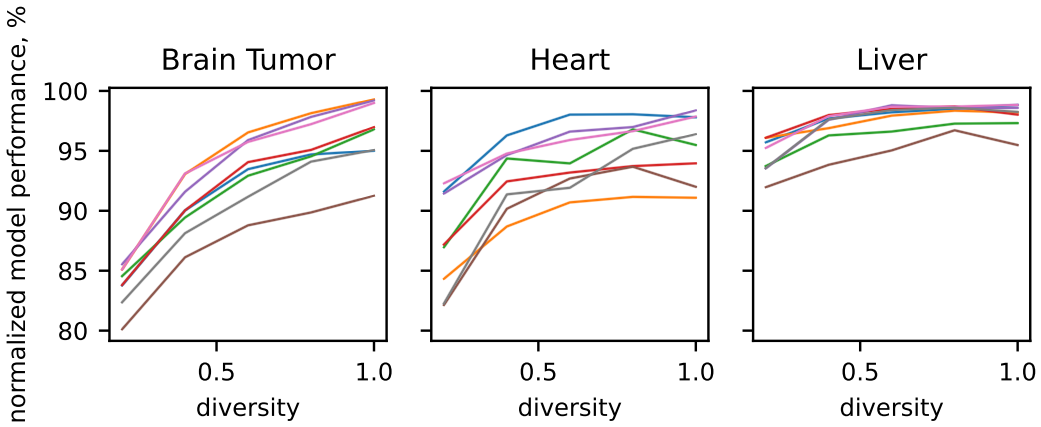



FIGURE 4.7: Plateauing of the model performance with the increase of the *diversity*. No matter how other virtues vary (different colors), saturation comes at the same point. This brings me to conclusion that one can track the model saturation w.r.t. the *diversity*, without fixing other parameters of the dataset.

segmenting more volumes should be preferred to segmenting more random variations and segmenting slices more densely/interpolating them.

### 4.3.4 How Much Data *Diversity* is Enough?

Although, how to define a reasonable limit to stop searching for *diversity*, and start increasing *completeness*? I plot the performance of the model w.r.t. *diversity* in Figure 4.7, each line represents a different set of virtues. Since the total amount of the data possibly available is unknown, I can not define a numerical limit. Instead, I note that all lines saturate ca. at the same point of increasing diversity. Hence I can define where to stop increasing *diversity* and start *increasing* completeness, by continuously updating a segmentation model, while expanding the dataset.

## 4.4 Discussion

In this chapter, I have compared the importance of different ways to spend labeling efforts and presented a way to optimize the segmentation labeling procedure. I minimized the effort required to obtain the model of a specific quality. In general, I conclude that *quality* is more important than *diversity*, which is more important than *completeness*. Based on my experiments, I propose the following procedure to minimize the effort during labeling volumetric data for segmentation:

1. Start with segmenting slices, without interpolation. Aim for maximal quality affordable without pixel hunting, at least 90%.

2. Decide on your time budget and distribute slices to segment as evenly through diverse volumes as possible. Though, keep in mind, that the structure of interest may impose a minimal slice number per volume to capture all parts of the structure.

3. Train a model as early in the process as possible. This allows, first, deciding which areas require more markup (by means of active learning, or just by an

expert assessment of predictions), and, second, recognizing the moment when model performance starts to saturate w.r.t. *diversity*.

4. After hitting the saturation w.r.t. the *diversity*, increase the *completeness* either by adding more volumes or by interpolating more slices to squeeze the last performance percent.

# Chapter 5

# Pre-Training for Segmentation via Slice Ordering

In the extensive discussions that took place in Chapter 2 and Chapter 4, it was highlighted that Neural Networks display a particular propensity towards both the quantity and the quality of training data. The more information they have access to, and the better the quality of this data, the better their performance. This is a double-edged sword; while it enhances the overall capability of these networks, it also presents a unique challenge as acquiring large amounts of high-quality data is not always feasible. In this chapter I propose another solution to the problem of the problem of the model's performance. This is a so-called pre-training technique, where the model is first trained on a set of the data without labels (the pre-training phase), and only then is tuned for the actual task at hand (fine-tuning phase). I present a novel algorithm of pre-training developed specifically for the tomographical data of the biological samples, in particular, Medaka fish.

## 5.1 Introduction

The typical response to the issue of the lack of the labels is to implement some form of pre-training, which essentially leverages openly available datasets or unlabeled samples from the very same training dataset in order to boost the initial training process. For instance, it is not uncommon for Neural Networks to be pre-trained on the ImageNet (Russakovsky et al., 2015) dataset, which is freely accessible and generally aids in improving outcomes for subsequent tasks, particularly those related to natural images.

However, while for openly available datasets, labels are readily provided, the task of training on unlabeled samples requires the development and implementation of novel approaches. One such approach involves self-supervised pre-training. This method hinges on the notion of learning from the data itself, allowing the model to develop representations without external guidance. While the idea of self-supervised learning is not new, the adaptation of this approach for specific applications is a nontrivial task.

In my work, I have specifically designed and developed an algorithm for self-supervised pre-training that is particularly suited to biomedical CT. The rationale behind this specialized approach was to mitigate the limitations often encountered in general pre-training strategies when applied to specific tasks like biomedical imaging.

The algorithm I developed is categorized as a data-driven prior, (see Chapter 2). Nonetheless, it deviates from some of the methodologies discussed therein, specifically those developed for medical data that assume perfect alignment. In contrast,

the devised approach is specifically designed to be resilient to imperfect alignment, a characteristic often encountered in practical applications. While most algorithms require perfect alignment, my approach only requires one axis where the ordering of the slices can be determined based on the sample of interest and not from the background content. This is particularly useful in real-world scenarios where obtaining perfectly aligned samples can be challenging and laborious.

For such datasets, I propose to train a model to predict the order within a batch of slices, randomly sampled from one volume. I refer to this method as *SortingLoss* hereinafter.

## 5.2 Method

The proposed *sorting loss* draws inspiration from the concept of the jigsaw puzzle task (Noroozi and Favaro, 2016). In the jigsaw puzzle task, an image is divided into several tiles which are then shuffled. The model is then tasked with predicting the relative positions of these shuffled tiles in order to reconstruct the original image.

However, *sorting loss* introduces a distinctive twist. Instead of predicting the relative positions of shuffled tiles, the model is now required to predict the order in a set of slices randomly extracted along one axis from a volume. In doing so, the model needs to learn the inherent structure within the data, much like understanding the overall picture in a jigsaw puzzle task, allowing it to predict the sequence of the slices effectively.

For the loss to enforce effective sample localization, the sample should exhibit less permutation invariance (along the slicing axis) than the background. The easier way to predict the order of slices would then be to learn the features of the sample and their relative positions. To illustrate, one could consider that the liver usually lies between the heart and the tail. By learning these relative positions of organs, the model can successfully predict the order of slices even when shuffled.

Considering the typical lack of positional alignment between different volumes in biological imaging, the calculation of loss is proposed to be performed only between slices obtained from the same volume. Not only this allows training on the misaligned volumes, but also makes a more reasonable and fair measure of the model's performance.

During the training phase, I construct a batch of size $k$ by randomly selecting one volume and subsequently randomly sampling a set of slices $\{x_i\}_1^k$ from this volume. Indices $\{\psi_i\}_1^k$ are uniformly sampled to represent each slice.

The model $f_\theta$ is trained with the loss outlined in Equation (5.1), which is a variation of the well-known *margin ranking loss* (Sculley, 2009). For each possible pair of predictions $f_\theta(x_i), f_\theta(x_j)$ where $\psi_i > \psi_j$, the loss is defined to enforce that $f_\theta(x_i) \geq f_\theta(x_j) + m$, with $m$ being a margin value set to prevent the collapse of representations. The process of training and the intuition behind the work of the model are depicted in Figure 5.1.

$$\mathcal{L} = \sum_{i,j \in [1,N]} [\max(0, (f_\theta(x_j) + m)) - f_\theta(x_i)]_{\psi_i > \psi_j} \tag{5.1}$$

In the absence of additional information, the slice indices $\psi_i$ could be uniformly sampled. However, there may be instances where the sample is known to have a high margin to borders. In these cases, it may be advantageous to use a generalized

Gaussian distribution, which discourages the sampling of slices that lack substantial content. This is proposed as a potential enhancement to improve the learning process of the model.

The generalized Gaussian distribution is a family of continuous probability distributions which includes the normal distribution as a specific case, with the probability density function described in Equation (5.2). It is parameterized by a location parameter $\mu$, which determines the center of the distribution; a scale parameter $\alpha > 0$, which influences the spread of the distribution; and a shape parameter $\beta > 0$, which can adjust the shape of the distribution from the symmetric bell curve of the normal distribution to heavy-tailed or light-tailed distributions. The $\Gamma$ stands for the gamma function.

$$f(x;\mu,\alpha,\beta) = \frac{\beta}{2\alpha\Gamma(\beta^{-1})} e^{-\left(\frac{|x-\mu|}{\alpha}\right)^{\beta}} \tag{5.2}$$

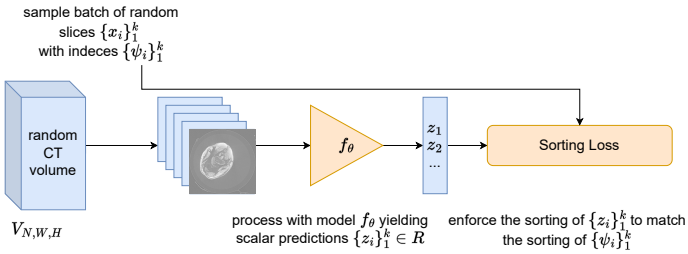
To provide a comprehensive measure of the model's performance, I propose the inclusion of a control metric alongside loss tracking. This metric is designed to offer a clear understanding of the model performance. The measure, called mean displacement, is defined as the mean absolute distance between the predicted order and the true order of slices. The mean displacement can be calculated as described in Equation (5.3). The intention behind this metric is to provide a human-interpretable measure of how well the model has learned to recognize the features of a sample rather than of a background.

$$\begin{aligned} \mathcal{L}_{MD}(\{psi_i\}_1^k, \{f\theta(x_i)\}_1^k) &= \frac{1}{k}\sum_i |\xi_i - \hat{\xi}i| \\ \xi &= argsort([\psi_1, ..., \psi_k]) \\ \hat{\xi} &= argsort([f\theta(x_1), ..., f_\theta(x_k)]) \end{aligned} \tag{5.3}$$

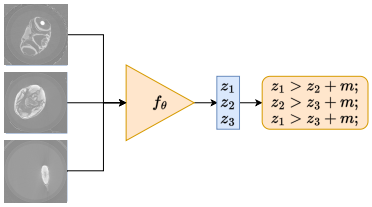
There are a couple of potential applications for the pre-trained model that interest us. The first involves using the pre-trained model as an encoder for a segmentation model. The second explores the possibility of employing the model for fully automated sample cropping. These two applications harness the capabilities of the model in different ways, exploiting its capacity to understand and predict the inherent structure within the data.

In the context of semantic segmentation, leveraging pre-trained encoders, such as ResNet, can significantly improve the performance of the model. The idea behind using a pre-trained encoder is to utilize the hierarchical feature extraction capabilities learned from large-scale datasets (like ImageNet), and adapt it to the task of segmentation. An encoder trained on a large and diverse dataset has learned to extract general, reusable features from raw data, which often are transferable to other tasks. In fact, using the encoder pre-trained on a large dataset became a de-facto standard in industry and a baseline to compare to in science.

One common architecture used for segmentation tasks is the U-Net model, which essentially consists of an encoder path and a decoder path. By using a pre-trained model like ResNet as the encoder, I can initialize the U-Net with a robust set of features. These features can then be fine-tuned to the specific task of segmentation, while the decoder part learns to map these high-level features to pixel-wise class probabilities. This combination of a pre-trained encoder with a specialized decoder often leads to superior results in comparison to training the whole model from




(A) Formation and training of one batch. A single CT volume $V$ is sampled randomly for each batch. After position indices are sampled as $\psi_i \sim \mathcal{U}(1, N)$, I take slices as $x_i = V_{\psi_i}$. I used pairwise margin ranking loss as the training objective.




(B) Intuitive example behind the *sorting loss* calculation. Being fed with three slices of head, body, and tail, the model should predict three ascending values. The model for this should remember the outlook of the sample, and its structure.

FIGURE 5.1: Scheme (a) and intuitive example (b) of the sorting loss training procedure.

scratch, especially when dealing with smaller or less diverse datasets. The strategy reduces the risk of overfitting and speeds up the convergence of training, resulting in a more efficient and robust model.

Using the model for automated cropping requires additional explanation and design of the own precedure. In this context, I hypothesize that since the model relies on the sample structures for prediction, these regions should display higher uncertainty when perturbed. The modifications introduced to the model to facilitate this are twofold.

Firstly, to estimate the uncertainty for different regions rather than on a global scale per image, the Global Average Pooling layer and the last linear layer are removed. This adjustment allows the model's output to be a spatial map of image region embeddings, akin to the Fully Convolutional Network (FCN) described in Chapter 2.

Secondly, the model is trained with dropout to estimate uncertainty. However, the inference with dropout does not follow the classical method; I permit dropout sampling during the inference time (Fabi and Schneider, 2020). This strategy allows sampling from the distributions of superpixel embeddings. To identify the region of interest, all superpixels where the standard deviation of the embedding exceeds a manually set threshold are selected. In this way, the regions that the model is relying on mostly (and thus, most likely contains the structures of interest) can be easily identified and cropped for further analysis.

## 5.3 Dataset

Model organisms are undeniably vital assets in diverse domains of scientific exploration, including but not limited to genetics, neuroscience, developmental biology, and pharmacological research. These organisms act as representative approximations of intricate biological systems, enabling the investigation of basic life processes in a manageable and simplified environment. In order to harness the full potential of these model organisms, the necessity for first-rate 3D atlases, rich in detail, becomes evident. These reference atlases provide expansive data concerning their development and anatomical structure. Such a reference framework enables the mapping of genes and other molecular markers onto distinct organs and tissues. This leads to a deeper comprehension of the interplay between phenotype and gene expression, as

well as the influence of gene regulation alterations on organism development and function.

Quality-rich 3D atlases have been created for numerous model organisms, ranging from zebrafish to mice, and fruit flies among others. One such example is the zebrafish atlas, which offers a meticulously detailed map of zebrafish brain structure, down to the molecular and cellular levels. This facilitates the study of brain function and development. In a similar fashion, the mouse atlas offers an all-inclusive perspective on mouse anatomy, and it is a tool frequently employed in mouse genetic studies and functional genomics research. Such 3D atlases act as pivotal tools for researchers, offering valuable insights into the fundamental biological processes and fostering novel discoveries that have significant implications for understanding human health and disease.

The Medaka fish, or Oryzias latipes, holds particular significance as a model organism due to a unique combination of biological and genetic characteristics that render it highly compatible with a multitude of research applications. These applications span genetics, functional genomics, and developmental biology. The Medaka fish boasts a relatively compact and thoroughly annotated genome, making it an optimal species for conducting genetic analysis and functional genomics studies. Another feature that adds to its appeal as a model organism is its ability to produce a large number of offspring within a limited time span. This trait is beneficial for both forward and reverse genetic screens. Additionally, the Medaka fish's transparent embryonic stage allows for direct visualization and manipulation of early developmental processes. It also possesses unique biological characteristics such as its simple and rapid lifecycle, adaptability to laboratory conditions, and compatibility with a diverse array of experimental techniques. These techniques include transgenesis, knockdown, and imaging, all of which enhance its appeal as a model organism for studying a broad spectrum of biological processes, from stem cell biology and embryonic development to neurodevelopment. The Medaka fish also exhibits high levels of genetic and phenotypic diversity, making it a compelling species for studies centered around evolutionary and population genetics.

The dataset contains CT scans of the Medaka (*Oryzias latipes*) fish scanned by the protocol alike Weinhardt et al., 2018. It contains in total 274 volumes that have an average size of $3000 \times 1008 \times 1008$ pixels. Since during the imaging, the helical CT was used to scan the full body of the fish, some changes in the size along the first axis are expected. 24 volumes were segmented by a panel of experts to contain labels of the visual system. Since we arranged the segmentation in such a way, that different experts had overlaps in their assigned volumes, we can estimate the expert quality of the segmentation. The Table 5.1 lists the segmented sub-organs and corresponding expert segmentation quality.

## 5.4 Experiments

### 5.4.1 *Medaka* Fish Segmentation

In the pursuit of generating a high-fidelity, segmented atlas of the Medaka fish, I decided to incorporate the concept of pre-training on the available data. This decision was guided by the potential of pre-training to reduce the necessity for extensive markup, thus providing a pragmatic basis for my investigation. The core idea is to leverage this approach in order to significantly enhance the efficiency and accuracy of the subsequent segmentation task, ultimately delivering a comprehensive, well-segmented atlas of the Medaka fish that would be invaluable for biological research.

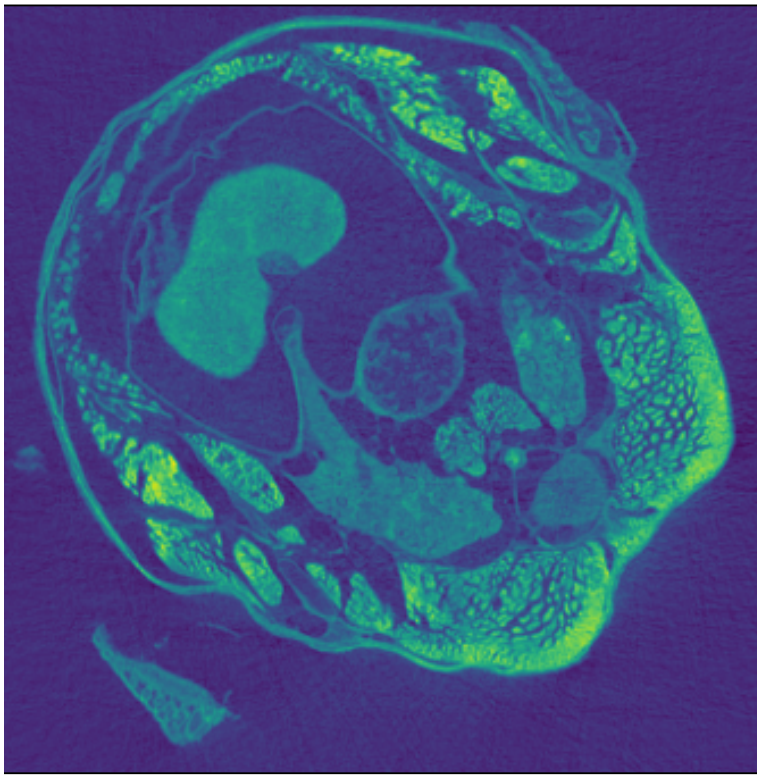


FIGURE 5.2: A slice example of the Medaka fish along the transverse plain.

| Sub-organ name | mIoU |
|---|---|
| left/right ciliary body and iris | 60.3 |
| left/right cornea | 12.1 |
| left/right lens | 92.6 |
| left/right muscles | 68.5 |
| optic nerve | 58.1 |
| left/right retina | 94.4 |
| average (without cornea) | 74.8 |

TABLE 5.1: The quality of the segmentation as provided by the experts. The quality was estimated on the samples independently segmented by different experts.

| data % | pre-training None | SimCLR | MoCoV3 | *sorting loss* |
|---|---|---|---|---|
| 1 | 63.5 | 66.6 | **67.9** | 67.4 |
| 100 | 74.8 | 72.7 | 74.5 | **75.0** |

TABLE 5.2: IoU of Medaka visual system segmentation, depending on the pre-training loss and amount of supervised dataset used. The median values are estimated on 5 independent runs.

To comprehensively assess the performance of this proposed pre-training methodology, I established a comparative framework featuring two popular data-prior pre-training methods: SimCLR (Chen et al., 2020b) and MoCoV3 (Chen, Xie, and He, 2021). The selection of these two methods was deliberate and informed by a dual rationale. Firstly, the paradigm of contrastive learning, which these methods embody, is a well-researched area that boasts open implementations thoroughly tested across a multitude of datasets. This advantage is significant as it mitigates the risk of performance degradation due to re-implementation errors, which can often accompany complex methodologies. Secondly, it was important to acknowledge that knowledge-driven pre-training methods typically make specific assumptions about the nature of the data (as discussed in Chapter 2).

These assumptions may not necessarily hold true for the dataset under consideration in this research. This realization further endorsed the choice of these two contrastive learning methods for comparison, as they are not reliant on specific data assumptions and therefore offer a more universally applicable solution. The eventual aim is to ascertain the most effective pre-training method for our specific task: the creation of a detailed, segmented atlas of the Medaka fish.

All of the selected pre-training algorithms were used on the same model class, ResNet-18, which was later used as an encoder for DeepLabV3+ models. For the pre-training, I randomly selected 50 unlabeled volumes and split them in 3-to-1 proportion to a train and a validation set. I trained a ResNet-18 model with the *sorting loss*, with the margin set to $m = 0.2$. All methods were trained for the same amount of epochs. However, since both contrastive learning methods require paired sampling, they have a larger computational footprint per epoch. I demonstrate the quantitative comparison of different pre-trainings in the Table 5.2. From the results, I conclude that the proposed method performs on par with the MoCoV3, while clearly outperforming SimCLR.

As Zoph et al., 2020 demonstrated, pre-training methods often decrease the quality compared to the baseline when lots of training data are available. The presented results indeed indicate that this is the case for the contrastive pre-training methods, while the proposed method doesn't demonstrate performance degradation.

However, the landscape of practical applications often requires the training of larger models. In recognition of this reality, I extended my investigation to examine the capacity of the proposed pre-training method to handle larger models and datasets. Consequently, I tried the pre-training of a larger, more complex architecture, specifically the ResNet-152 model, utilizing all 250 unlabeled volumes available in the dataset.

Due to constraints related to computational resources, I found myself unable to conduct a comparison with other methodologies. Despite this limitation, the outcomes of this extensive pre-training exercise were revealing. Notably, even when the U-Net was subjected to fine-tuning, the pre-training process still managed to improve performance. In terms of the Intersection over Union (IoU) metric, there was

a 2.3% performance enhancement compared to the training of a model of identical size from scratch. This gain was accomplished while utilizing the full extent of the supervised datasets, underlining the effectiveness of the pre-training process even in larger-scale application scenarios.

This outcome reaffirms the value of the pre-training approach in improving model performance, even when handling more complex model architectures and larger datasets. It lends strength to the broader applicability of this approach in developing comprehensive, high-fidelity 3D atlases for model organisms, such as the Medaka fish, thereby facilitating more nuanced and detailed biological research.

**NB!** During the experiments, I noted that the aggressive augmentation can further increase quality for the *sorting loss* pre-training while decreasing it for the SimCLR. This finding completely agrees with the recent findings by Wang et al., 2021, who states that aggressive augmentation may damage the performance of the contrastive pre-training methods.

### 5.4.2 *Medaka* Fish Localization

As it was noted in the Section 5.2, I can use the pre-trained model to localize the sample. In the first experiment, I consider the qualitative examination of the usage of a pre-trained model for sample localization. For the pre-training, I selected 50 unlabeled volumes and randomly split them in 3 to 1 proportion to a train and a validation set. I trained a ResNet-18 model with the *sorting loss*, with the margin set to $m = 0.2$. The mean displacement reached 0.42 on the test set and 0.26 on the train set for the batch size of 12. Hence, I conclude that the estimator provides reasonably good ordering and should rely on the data structures.

To define the bounding box around the sample, I followed the training procedure, described in the Section 5.2, with the dropout layer added after the first, second, and third residual blocks. To define the bounding box I used 3% of the superpixels with the highest standard deviation. Additionally, I found it beneficial to use only the largest connected region of the pixels above this threshold.

As a baseline, I select pixel value thresholding, since it is still a frequently used technique. I used the Otsu threshold finding method. To compare the results numerically, I selected 20 volumes, that were not included in the training set, performed the Otsu thresholding on them, and corrected it by hand, to match the expected bounding box.

Direct numerical comparison of the thresholding versus the NN localization favors thresholding with the mean bounding box IoU of the thresholding being 0.87, while 0.47 for the sorting loss localization. To further assess the quality, I consider four types of errors sorted from the most to least severe: `fish not inside`, `cut fish parts`, `cut fins`, and `oversize bounding box`. The results for the 20 volumes are presented in the Table 5.3. So, the most popular problems for localization with sorting loss are cutting the fins and overshooting the boundaries. The different types of errors are presented in the Figure 5.3.

Based on this experiment, this localization method could be recommended for cases, where either it is problematic to employ the classical localization methods (e.g., due to the low contrast between the sample and the background), or where the robustness of the prediction is more important than the tightness of the bounding box. For example, to adjust the reconstruction parameters, it is useful to know the position of the imaged sample, and for this, the proposed localization method provides sufficient performance. Also, it's worth noting, that even after adding safety

| error type | neural network | Otsu threshold |
|---|---|---|
| `fish not inside` | 0 | 2 |
| `cut fish parts` | 4 | 0 |
| `cut fins` | 18 | 0 |
| `oversize` | 12 | 0 |

TABLE 5.3: The quantitative comparison of the bounding box errors between the proposed method and the Otsu thresholding. The errors are sorted top-down from the most to least severe. While the proposed method typically fails to produce a tight bounding box, it is more robust and less tuned to the exact gray values.

borders around the bounding box provided by the model, so that it doesn't cut the important parts off, it was able to reduce the size of the volume by 74% per volume on average, i.e. from 24 Gb to 5 Gb.

## 5.5 Discussion

The pre-training approach presented in this study can be described as occupying a distinct middle ground between the broad, data-prior pre-training methods and the task-specific knowledge-prior pre-training approaches typically employed in the medical domain. In contrast to the more general, data-prior pre-training methods, which require no assumptions about the data but demand significant computational resources, the method proposed herein offers a more balanced approach.

On one hand, the proposed method leverages task-specific information to optimize pre-training effectiveness without relying on an overabundance of computational resources. Yet, it doesn't over-specify to the point of being incompatible with more general types of data, as is often the case with some knowledge-prior pre-training methods, for instance, those that assume perfect data alignment. Therefore, the method carves out a novel, middle-ground position in the pre-training landscape that balances resources and performance, generalization, and specificity.

What is striking about this balance is the performance of the model relative to the computational resources employed. Despite being less resource-intensive, the model competes favorably with modern contrastive pre-training methods in the downstream task of semantic segmentation. Even when the downstream task is trained in a full-data regime, it does not negatively impact the model performance, an outcome that underscores the model's efficiency and robustness.

However, the method's performance in terms of bounding box IoU, when applied to localization, is inferior compared to a simple baseline. Despite this, it demonstrates increased robustness in scenarios where the voxel's gray value is insufficient to make accurate determinations. In these challenging situations, the model's ability to maintain performance underscores its value in real-world applications where the data often contain unexpected and complex patterns. While the bounding box IoU score may be lower, the model's overall performance in context-sensitive situations suggests an exciting direction for future research and refinement of this pre-training approach.

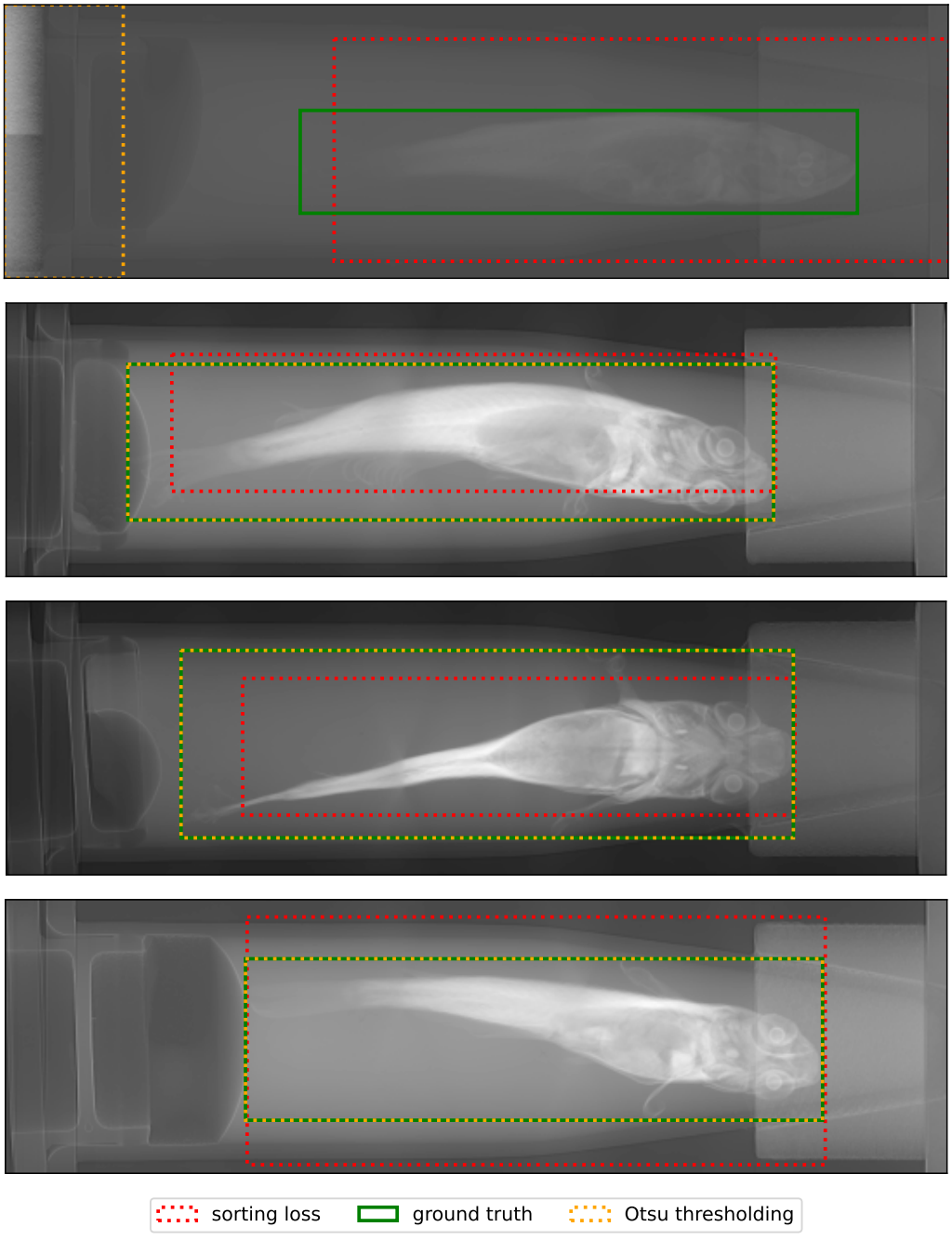



FIGURE 5.3: Examples of the bounding box errors, depicted top-down from the most to least severe. (1) Otsu thresholding fails to localize the fish in presence of an artifact. (2) the proposed method fails to include the entire fish, cutting off important parts of the head. (3) the proposed method cuts off the fins and tail while preserving all important parts of the sample. (4) the proposed method fails to produce a tight bounding box.

# Chapter 6

# Self-Training for the Medaka Fish Segmentation

The work presented in this chapter is strongly related to the (Bhatt et al., 2023), which was published in the proceedings of the ISBI conference. The work was performed in collaboration with *Jwalin Bhatt* conducted experiments for this work.

Morphological atlases hold a vital position as a significant tool in the study of organisms and their structure. They provide an exhaustive, detailed visual representation of the organism's structure, enabling in-depth comparative studies. The advent of modern high-throughput Computed Tomography (CT) facilities has revolutionized this space, offering the capability to generate hundreds of full-body, high-resolution volumetric images of organisms. This transformation marks a significant stride in organismal studies, allowing researchers to delve deeper into the organism's structural nuances with unprecedented precision.

Despite these advances, for these volumetric representations to mature into a comprehensive morphological atlas, they necessitate meticulous organ segmentation. Organ segmentation is the process of identifying and delineating different organs within the volumetric images, thereby creating a detailed map of the organism's internal structure. This segmentation process is crucial to maximize the usefulness of the CT-generated images, providing the necessary detail and clarity for further studies.

Over the past decade, machine learning techniques have emerged as a game-changer, achieving remarkable feats in image segmentation tasks. Machine learning algorithms, with their ability to learn complex patterns and relationships from data, have proven incredibly effective at identifying and separating distinct organs within the volumetric images. These techniques have not only enhanced the speed and accuracy of segmentation but have also reduced the reliance on manual, labor-intensive processes.

However, these machine learning techniques, especially deep learning methods, are data-hungry entities, necessitating ample annotated data to be trained effectively. The requirement for such vast quantities of annotated data poses a substantial challenge, especially considering the complexity and detailed nature of the 3D segmentation process.

In light of these challenges, this paper puts forth an innovative solution in the form of a self-training framework for the multi-organ segmentation of Medaka fish

in tomographic images. The proposed method seeks to circumvent the limitations of traditional segmentation techniques while harnessing the power of machine learning, all without the need for copious amounts of annotated data.

The proposed approach leans heavily on the concept of pseudo-labeled data generated by a pretrained 'Teacher' model. In the realm of machine learning, pseudo-labeling is a semi-supervised learning method where a model is initially trained on a small set of labeled data, and this model is then used to predict labels for the unlabeled data, thereby creating pseudo-labels. The pseudo-labeled data from the 'Teacher' model forms the backbone of the proposed framework.

The pseudo-labeled data is further refined by adopting a Quality Classifier. The Quality Classifier, as the name suggests, is tasked with assessing and improving the quality of the pseudo-labels, thereby enhancing the reliability of the data being fed into the model for training. This data refinement process is instrumental in ensuring the robustness and accuracy of the final segmentation model.

Furthermore, I introduce a pixel-wise knowledge distillation method. Knowledge distillation is a process where the learned representations (knowledge) of one model (usually a larger, more complex model) are transferred to another model (usually smaller and simpler). The pixel-wise knowledge distillation method is implemented to prevent overfitting to the pseudo-labeled data and bolster the segmentation performance. By mitigating overfitting, this method ensures that the model is more generalizable and better equipped to handle new, unseen data.

The experimental outcomes demonstrate that the proposed method results in a notable improvement of 5.9

## 6.1 Introduction

The study of model organisms has a long history. Initially, most of the studies were done either on dissection or by visual observation of the transparent organisms (e.g., Zebrafish). In recent years, the growing capacity to produce radiographic images led to qualitative change in the available data. Synchrotron-based micro Computed Tomography allows the production of images of high resolution (micrometer scale pixel sizes) (Sombke et al., 2015). And with further automatization of the scanning process, tens of volumes could be obtained serially, without human interaction.

Fishes have become increasingly important model organisms in biomedical research over the past decades. In particular, the Medaka (Oryzias latipes) has become an indispensable model organism for studying gene function in vertebrates. Their availability for future-oriented genetics, in particular their small body size, the transparency of their embryos, and extra-uterine development, make them ideal systems for systematic investigations of developmental processes. The digital morphological atlas of the adult Medaka fish allows biologists to analyze the morphometric properties of internal and external features, including organs and tissues. Therefore, comparative studies concerning phenotypic analysis use the atlas as a reference and are thus of vital importance for the further use of Medaka as a model organism. The quantitative description of the organ positions and shapes can be discovered by solving the segmentation task. Although, the study of a bigger number of specimen samples faces the hurdle of semi-manual segmentation, which consumes lots of time.

The conventional technique used to generate the 3D anatomical atlas for the Medaka fish was using the Amira software with the help of the readily available

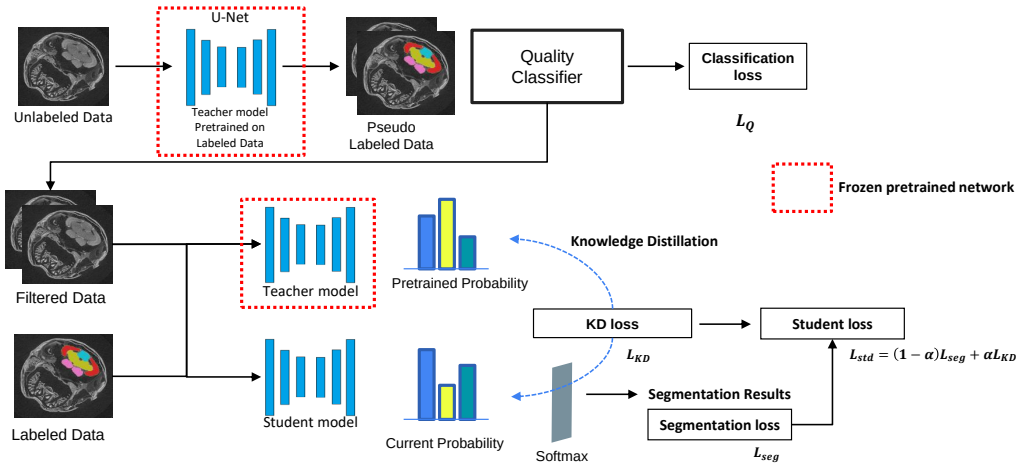



FIGURE 6.1: Overview of the proposed framework. The teacher model is trained on the labeled data and the pseudo-labeled data are obtained by the trained teacher model. The quality classifier refines pseudo-labeled data and the student model is trained on the filtered pseudo-labeled data and labeled data by using knowledge distillation to improve the performance of the organ segmentation.

annotations and atlases (Kinoshita et al., 2009; Shanthanagouda et al., 2014; Bryson-Richardson et al., 2007). This, however, required an immense amount of hand work even for small data sets and lacks automatization for scaling to larger datasets. To handle these limitations, an atlas-based approach was proposed by Weinhardt et al., 2018, which allows the automatic segmenting of new samples.

However, the atlas-based methods may suffer from quality loss, given the strong morphological differences between the new sample and the base segmentation.

Recently, deep learning-based methods so-called semi-supervised learning have been introduced to solve the problems with the cost of requiring way a large amount of labeled data. In Shen et al., 2019, a weakly supervised segmentation method using bounding boxes instead of segmentation masks was proposed to reduce the cost of labeling data. In addition, the pre-training technique aims to find a good, data-driven initialization for the model weights. He et al., 2020 proposed an object segmentation framework, called Mask R-CNN, which improves the segmentation performance by pre-training the neural network model on a large open-source dataset, such as ImageNet, and training the model on a small private dataset again. The self-training technique, in contrast, employs the unlabeled part of the dataset (Hsu et al., 2019). The core idea is to train two models, the Teacher and the Student. The Teacher model (which could be an ensemble of models) is trained on the small labeled dataset. Subsequently, its predictions on the unlabeled part of the dataset (called pseudo-labels) are used to train the Student model. To further improve the performance of self-training, several methods have been proposed. In Tarvainen and Valpola, 2017, Mean Teacher, which averages model weights to provide a better Teacher model, was proposed. Xie et al., 2020 proposed NoisyStudent which adds noise to the Student to transfer knowledge from the Teacher's knowledge. Zou et al., 2018b introduced a class-balanced self-training to select the pseudo-labels better to use for training the Student.

In this chapter, I propose a knowledge distillation framework for Medaka organ segmentation in tomographic images with pseudo-label refinement. I utilize the pseudo-labeled data from the Teacher model pretrained on the labeled data and

adopt a Quality Classifier that learns to distinguish well-segmented data and bad-quality segmented data. The Student model is trained on the labeled data and the refined pseudo-labeled data to improve the segmentation performance. To prevent overfitting to the pseudo-labeled data and further improve the performance, I introduce a pixel-wise knowledge distillation technique that regularizes the Student to learn predictive capability from the output segmentation prediction of the Teacher explicitly. The proposed method is evaluated on the Medaka tomographic image dataset I have collected, and the experimental results show that the method improves the Medaka segmentation performance effectively.

## 6.2 Proposed Method

I show the proposed framework in fig. 6.1. It consists of three models: (1) the Teacher model, trained on the labeled data, (2) the Quality Classifier trained to distinguish bad pseudo-labels from good, and (3) the Student model, trained on the filtered pseudo-labeled data together with the labeled data. I further describe components of this framework in detail.

### 6.2.1 Models

First, I use a U-Net (Ronneberger, Fischer, and Brox, 2015) for the **Teacher model** and train on the data with their corresponding segmentation provided by the biologists. U-Net introduces skip concatenation between the encoder and the decoder layers and provides good performance in image segmentation. Let $X_L = \{x_L^i\}_{i=1}^{N_L}$ be the labeled input images with pixel-wise annotated labels $Y_L = \{y_L^i\}_{i=1}^{N_L}$. In this work, I adopted ResNet-18 (He et al., 2016b) as the encoder to achieve high performance for image classification. Then, I generate pseudo-labeled data by putting all the unlabeled data into the teacher model. I denote the unlabeled images $X_U = \{x_U^i\}_{i=1}^{N_U}$ and the teacher model $F_T : \mathcal{X}_\mathcal{U} \to \mathcal{Y}_\mathcal{U}$. Then, the pseudo labels can be defined as $Y_U = F_T(X_U) = \{y_U^i\}_{i=1}^{N_U}$.

Among these pseudo labels, there are many slices that are harmful to be used for training the Student model. In this work, I adopt a **Quality Classifier** to distinguish between good and bad pseudo labels. To train the quality classifier, a dataset of about 1000 slices was labeled manually into good and bad categories. The slice from the image was concatenated with the mask, where each sub-organ in the mask was represented using a separate channel. Thus, the refined input images are expressed as $X_R = \{x_R^i\}_{i=1}^{N_R}$ with the refined pseudo labels $Y_R = \{y_R^i\}_{i=1}^{N_R}$.

For the **Student model** use the same architecture and size of the model as the Teacher. As shown in fig. 6.1, the Student is trained on the labeled data and filtered data by the quality classifier to prevent performance degradation due to the inaccurate pseudo-labels. However, the number of refined pseudo-labeled data is much more than the number of labeled data, so the Student can be biased to the pseudo-labeled data. I concatenate the refined pseudo-labeled data and the labeled data $X_C = \{x_c^i\}_{i=1}^{N_L+N_R} = X_L \cup X_R$ with the labels $Y_C = \{y_c^i\}_{i=1}^{N_L+N_R} = Y_L \cup Y_R$. I denote the Student model $F_T : \mathcal{X}_\mathcal{C} \to \mathcal{Y}_\mathcal{C}$. Then, I can express the predictive segmentation maps from the Teacher and the Student model as $\tilde{Y}_C^T = F_T(X_C)$ and $\tilde{Y}_C^S = F_S(X_C)$, respectively.

### 6.2.2 Training

I define the segmentation loss on the combination of the pseudo-labeled and supervised data as follows.

$$\begin{aligned} L_{seg} &= L_{CE}(F_S(X_C), Y_C), \\ \text{where } L_{CE} &= -\sum_k q(k)\log(p(k)), \end{aligned} \tag{6.1}$$

I train the Student with this loss drives to transfer the knowledge from the Teacher model. Hereinafter, I refer to training with this loss solely as **Pseudo-Labeling**.

I incorporate the idea of the **Self-Training** (Hsu et al., 2019) by initializing the Student model with the last checkpoint of the Teacher model. This is supposed to lead to better convergence.

Furthermore, I incorporate the idea of the **Knowledge Distillation** (Hinton, Vinyals, and Dean, 2015; Yuan et al., 2020), also known as dark knowledge distillation. To prevent overfitting to the pseudo-labeled data and keep the knowledge from the labeled data in the pretrained Teacher model, this idea proposes directly distilling the softened labels produced by the Teacher Model. As proposed by the listed papers, I use temperature scaling to soften the predictions of the Teacher model.

$$p_k^t(x^i;\tau) = softmax(z_k^t(x^i;\tau)) = \frac{\exp\left(z_k^t(x^i)/\tau\right)}{\sum_j^K \exp\left(z_j^t(x^i)/\tau\right)} \tag{6.2}$$

where $p_k^t(x^i;\tau)$ is the $k$-th output of $i$-th pixel, $K$ is the number of segmentation classes, $z_k^t$ is the pixel-wise output segmentation logits of the pre-trained teacher model and $\tau$ is the temperature to soften the predictive segmentation probability. Using the softened predictions, I regularize the student model by using the Kullback-leibler (KL) divergence for all pixel pairs at the same spatial position with the teacher model. The knowledge distillation (KD) loss is given as follows.

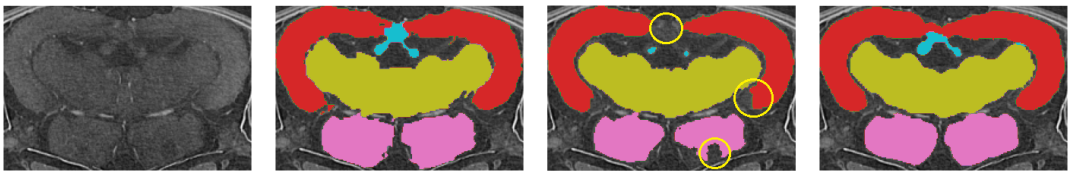
$$L_{KD} = \frac{1}{N}\sum_{i\in N} KL(p^s(x^i;\tau) \parallel p^t(x^i;\tau)) \tag{6.3}$$

where $N = W \times H$ is the number of pixels of the image data, $KL(\dot{})$ is the KL divergence function, and $p^s(x^i;\tau)$, $p^t(x^i;\tau)$ are the output probability of the $i$-th pixel in the segmentation map from the student and the pre-trained teacher models respectively.

The proposed method combines all mentioned parts together: I use the pseudo-labels further filtered by the Quality Classifier, initialize the Student with the last Teacher model snapshot, and regularize the Student model via the Knowledge Distillation loss. The final student loss, with which the Student is trained, is defined as follows.

$$L_{std} = (1-\alpha)L_{seg} + \alpha L_{KD} \tag{6.4}$$

where $\alpha$ controls the relative importance of different losses.



(A) (B) (C) (D)

FIGURE 6.2: Examples of (a) Input image, (b) ground truth, and segmentation results by (c) the Teacher (Fully Supervised), (d) the Student model by the proposed method.

TABLE 6.1: Comparison of the segmentation results between the Fully Supervised model as a baseline, and the proposed method.

| Number of labeled volumes | mIoU (%) | | Dice (%) | |
|---|---|---|---|---|
| | proposed | baseline | proposed | baseline |
| 2 | 72.5 | 43.2 | 77.9 | 48.5 |
| 7 | 75.6 | 54.7 | 80.8 | 55.9 |
| 12 | 79.8 | 60.1 | 86.9 | 65.2 |
| 23 | 82.4 | 74.8 | 89.4 | 82.2 |

## 6.3 Experimental Results

In this section, I address a brief introduction of the tomographic Medaka image data and implementation details. In addition, I provide the comparison results and ablation study of the proposed method in terms of the mean intersection over union (mIoU) and the dice score.

### 6.3.1 Datasets

The data was collected several beam times, following the protocol proposed by Weinhardt et al., 2018. The original slices, the dimension of $6000 \times 2000 \times 2000$, were rescaled to $3000 \times 1000 \times 1000$. The labeled dataset used to train the teacher model consists of 30 samples. The unlabeled dataset consists of a total of 582 scans from three different experiments: 169, 232, and 181 samples The labeled data were split into the train (75%) and validation (25%), hence 23 volumes were used for training, and 7 were used for validation.

### 6.3.2 Implementation Details

The proposed method was implemented in Python with PyTorch library. I used two NVIDIA RTX 8000 and two RTX 2080 for training and testing. Adam optimizer was used to train the proposed network with a learning rate of 3e-4. The hyperparameters in eq. (6.4) were $\alpha = 0.1$ and $\tau = 4$. The models were trained with a crop size of 256 and a batch size of 64, and they converged within 15 epochs. The convergence was even faster (5 epochs) when initialized with the Teacher's weights.

### 6.3.3 Baseline Comparison

First, I quantitatively compare the results of the proposed method with the Fully Supervised model as a baseline. I present the comparison of the results in table 6.1. The proposed method clearly outperforms the baseline both in the low and full data regimes. It demonstrates that the method consistently outperformed the baseline.

TABLE 6.2: Evaluation results of ablation study for the proposed framework with knowledge distillation and quality classifier using 23 labeled volumes for training.

| Method | mIoU (%) | Dice (%) |
|---|---|---|
| Fully supervised | 74.8 | 82.2 |
| Pseudo-Labeling | 76.5 | 86.0 |
| Knowledge Distillation | 76.7(+0.2) | 86.6(+0.6) |
| Teacher Checkpoint | 77.5(+1.0) | 86.8(+0.8) |
| Quality Classifier | 81.2(+4.7) | 87.4(+1.4) |
| **Proposed** | 82.4(+5.9) | 89.4(+3.4) |

In addition, the proposed method using seven labeled volumes also provided better performance than the baseline using all available training data volumes, confirming the proposed method can improve the Medaka segmentation performance. Even the method using only two volumes of the labeled data gave better results than the Fully Supervised method with 12 labeled volumes.

Second, I visually compare the results of the segmentation as shown in fig. 6.2. I marked the problematic areas of the sample with yellow circles on fig. 6.2 (c). While the prediction of the proposed method clearly has its own peculiarities of segmentation, the provided result is smoother spatially and closer to the ground truth.

### 6.3.4 Ablation Study

I present the ablation study of the employed components in table 6.2. The proposed method performed slightly better than any of the parts of the pipeline separately. I note, that the proposed way of filtering the pseudo-labels is the most beneficial part of the improvements. I hypothesize, that training better Quality Classifiers with more data could be beneficial for the final quality, but leave a thorough assessment of this question for future research.

Interestingly, the results demonstrate that the improvements of the proposed method are equal to or greater than the sum of improvements provided by the separate components. This could mean, that different components improve the quality of the result in different ways. This is beneficial for total improvement since improvements of the separate components do not interfere with others.

## 6.4 Discussion

In this chapter, I have presented a knowledge distillation framework that improved the segmentation quality for the organs of the Medaka fish with pseudo-label refinement. The proposed method improved 5.9% of the mIoU and 3.4% of the Dice, measured in the full data regime, compared to the fully supervised training method. Notably, in the low data regime, presented with only 2 volumes as a training set, the method yields an improvement of 29.3% mIoU, and a result, which is on par with supervised training on 12 volumes. This helps to reduce the burden of the hand-drawn segmentation, excessively used by the deep learning models to be trained. I also proposed my own view on the selection of the pseudo-labels used to train the Student model, called Quality Classifier. This component is a core component of the proposed method and provides a noticeable contribution to the quality improvement (+4.7% mIoU out of +5.9% in total). In future work, I will collect more

tomographic data on Medaka and other types of fish with more organisms to evaluate and improve the proposed method. I believe extending this work by including more organisms can provide more understanding for future-oriented genetics.

# Chapter 7

# Conclusion

In the broader context of the rapidly growing data landscape and the need for robust analysis strategies, this doctoral thesis has explored the profound power and potential of advanced computer vision, with a particular focus on tomographic data. It brings to the fore, the benefits of incorporating machine learning models, especially deep learning algorithms, into the analysis of CT scans in various domains, including biology, medicine, and material sciences. This journey, however, is not without its hurdles, the most prominent of which include the need for large, accurately labeled datasets and domain-specific knowledge during model training. The thesis, through the presentation of four individual but closely related studies, presented solutions to these challenges from data pre-processing down to neural networks training.

Chapter 3 delved into the application of the Noise2Noise self-supervised denoising approach to multi-channel imaging datasets. This method appeared to be a robust and efficient alternative to conventional denoising methods and regularized iterative reconstruction methods. The method streamlined the denoising process by eliminating the need for manual parameter fine-tuning or regularization formulation and tuning, demonstrating significant improvements in image quality.

Chapter 4 of this thesis takes on the issue of data labeling, which is often the most labor-intensive aspect of model training. Through an extensive investigation into the aspects of data segmentation labeling, we drew conclusions on the relative importance of quality, diversity, and completeness with respect to the efforts spent on labeling. Drawing from these experimental results I proposed a procedure of the markup creation, which minimizes the practical effort.

Chapter 5 moves forward on this path by introducing the *SortingLoss* method, a resource-efficient self-supervised pre-training technique. This technique leverages the inherent order of slices in a CT scan volume to pre-train the neural network. Positioned between general and task-specific pre-training techniques, the SortingLoss method encapsulates the advantages of both. It demonstrated considerable performance, matching that of modern contrastive pre-training methods, in semantic segmentation tasks even when tested in a full-data regime. Furthermore, the method exhibited the ability to robustly perform sample localization without any supervision.

In Chapter 6, this work deepens the training approaches with the usage of the self-training techniques applied to multi-label segmentation. The chapter extended the well known pseudo-labeled data method with a novel Quality Classifier and a pixel-wise knowledge distillation technique. The case study is presented on the Medaka fish brain areas segmentation. The proposed approach led to a marked improvement in segmentation performance, particularly in a low-data regime, therefore, significantly reducing the need for hand-drawn segmentation.

In sum, during my journey, I explored the potential of integrating advanced machine learning techniques, particularly deep learning algorithms with relaxed supervision, in the domain of CT imaging analysis. In this work I presented a set of my findings spanning from pre-processing the acquired data to get better signal-to-noise ratio, to label-efficient model training. Each of these findings serves as a stepping stone toward the broader goal of making deep learning models more efficient, accessible, and applicable to a wide range of real-world problems. And along these, solving particular practical problems present in day-to-day life of a researcher in a CT lab.

# Bibliography


Ametova, Evelina et al. (May 2021a). *Code to reproduce results of "Crystalline phase discriminating neutron tomography using advanced reconstruction methods"*. Version 0.1. DOI: 10.5281/zenodo.4884710. URL: https://doi.org/10.5281/zenodo.4884710.

Ametova, Evelina et al. (2021b). "Crystalline phase discriminating neutron tomography using advanced reconstruction methods". In: *Journal of Physics D: Applied Physics* 54.32, p. 325502.

Antonelli, Michela et al. (June 2022). "The Medical Segmentation Decathlon". In: *Nature Communications* 13.1. ISSN: 20411723. DOI: 10.1038/s41467-022-30695-9. URL: http://arxiv.org/abs/2106.05735.

Avetisian, Manvel et al. (Mar. 2020). "Radiologist-level stroke classification on non-contrast CT scans with Deep U-Net". In: DOI: 10.1007/978-3-030-32248-9{_}91. URL: http://arxiv.org/abs/2003.14287.

Bardes, Adrien, Jean Ponce, and Yann LeCun (May 2021). "VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning". In: URL: http://arxiv.org/abs/2105.04906.

Batson, Joshua and Loic Royer (Jan. 2019). "Noise2Seif: Blind denoising by self-supervision". In: *36th International Conference on Machine Learning, ICML 2019* 2019-June, pp. 826–835. URL: http://arxiv.org/abs/1901.11365.

Bentley, Phillip M (Apr. 2020). "Instrument suite cost optimisation in a science megaproject". In: *Journal of Physics Communications* 4.4, p. 045014. ISSN: 2399-6528. DOI: 10.1088/2399-6528/ab8a06.

Bhatt, Jwalin et al. (Apr. 2023). "A Knowledge Distillation Framework for Multi-Organ Segmentation of Medaka Fish in Tomographic Image". In: *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)*. IEEE, pp. 1–5. ISBN: 978-1-6654-7358-3. DOI: 10.1109/ISBI53787.2023.10230689.

Bin Saeedan, Mnahi et al. (Aug. 2016). "Thyroid computed tomography imaging: pictorial review of variable pathologies". In: *Insights into Imaging* 7.4, pp. 601–617. ISSN: 1869-4101. DOI: 10.1007/s13244-016-0506-5. URL: http://link.springer.com/10.1007/s13244-016-0506-5.

Bishop, Christopher M. (Jan. 2007). *Pattern Recognition and Machine Learning*. Springer-Verlag. ISBN: 0387310738. DOI: 10.1117/1.2819119. URL: http://electronicimaging.spiedigitallibrary.org/article.aspx?doi=10.1117/1.2819119.

Boin, Mirko (2012). "NXS: a program library for neutron cross section calculations". In: *Journal of Applied Crystallography* 45.3, pp. 603–607.

Bradley, Robert S., Ian K. Robinson, and Mohammed Yusuf (Feb. 2017). "3D X-Ray Nanotomography of Cells Grown on Electrospun Scaffolds". In: *Macromolecular Bioscience* 17.2, p. 1600236. ISSN: 16165195. DOI: 10.1002/mabi.201600236. URL: https://onlinelibrary.wiley.com/doi/10.1002/mabi.201600236.

Bredies, Kristian, Karl Kunisch, and Thomas Pock (2010). "Total generalized variation". In: *SIAM Journal on Imaging Sciences* 3.3, pp. 492–526.

Bryson-Richardson, Robert J et al. (Dec. 2007). "FishNet: an online database of zebrafish anatomy". In: *BMC Biology* 5.1, p. 34. ISSN: 1741-7007. DOI: 10.1186/1741-7007-5-34.

Buades, Antoni, Bartomeu Coll, and Jean Michel Morel (2005). "A non-local algorithm for image denoising". In: *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*. Vol. II. IEEE, pp. 60–65. ISBN: 0769523722. DOI: 10.1109/CVPR.2005.38. URL: http://ieeexplore.ieee.org/document/1467423/.

Burca, G et al. (2013). "Modelling of an imaging beamline at the ISIS pulsed neutron source". In: *Journal of Instrumentation* 8.10, P10001.

Camattari, Riccardo et al. (Apr. 2020). "X-ray characterization of self-standing bent Si crystal plates for Large Hadron Collider beam extraction". In: *Journal of Applied Crystallography* 53.2, pp. 486–493. ISSN: 16005767. DOI: 10.1107/S1600576720002800. URL: https://scripts.iucr.org/cgi-bin/paper?S1600576720002800.

Campbell, Stuart I. et al. (Mar. 2021). "Outlook for artificial intelligence and machine learning at the NSLS-II". In: *Machine Learning: Science and Technology* 2.1, p. 013001. ISSN: 26322153. DOI: 10.1088/2632-2153/abbd4e. URL: https://iopscience.iop.org/article/10.1088/2632-2153/abbd4e.

Chen, Ting et al. (Feb. 2020a). "A simple framework for contrastive learning of visual representations". In: *37th International Conference on Machine Learning, ICML 2020*, pp. 1575–1585. URL: http://arxiv.org/abs/2002.05709.

— (Feb. 2020b). "A Simple Framework for Contrastive Learning of Visual Representations". In: URL: http://arxiv.org/abs/2002.05709.

Chen, Xinlei, Saining Xie, and Kaiming He (Apr. 2021). "An Empirical Study of Training Self-Supervised Vision Transformers". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9620–9629. ISSN: 15505499. DOI: 10.1109/ICCV48922.2021.00950. URL: http://arxiv.org/abs/2104.02057.

Dalsasso, Emanuele, Loic Denis, and Florence Tupin (Oct. 2022). "As if by Magic: Self-Supervised Training of Deep Despeckling Networks with MERLIN". In: *IEEE Transactions on Geoscience and Remote Sensing* 60. ISSN: 15580644. DOI: 10.1109/TGRS.2021.3128621. URL: http://arxiv.org/abs/2110.13148http://dx.doi.org/10.1109/TGRS.2021.3128621.

Davis, Graham, Nitin Jain, and James Elliott (2008). "A modelling approach to beam hardening correction". In: *Developments in X-ray Tomography VI*. Vol. 7078. SPIE, pp. 423–432.

Egan, CK et al. (2015). "3D chemical imaging in the laboratory by hyperspectral X-ray computed tomography". In: *Scientific reports* 5.1, pp. 1–9.

Evennett, P. J. and C. Hammond (2004). "Microscopy - Overview". In: *Encyclopedia of Analytical Science: Second Edition*. Elsevier, pp. 32–41. ISBN: 9780123693976. DOI: 10.1016/B0-12-369397-7/00376-9. URL: https://linkinghub.elsevier.com/retrieve/pii/B0123693977003769.

Fabi, Kai and Jonas Schneider (Aug. 2020). "On Feature Relevance Uncertainty: A Monte Carlo Dropout Sampling Approach". In: *arXiv*. DOI: 10.5281/zenodo.3970396. URL: http://arxiv.org/abs/2008.01468%0Ahttp://dx.doi.org/10.5281/zenodo.3970396.

Fan, Linwei et al. (Dec. 2019). "Brief review of image denoising techniques". In: *Visual Computing for Industry, Biomedicine, and Art* 2.1, p. 7. ISSN: 2524-4442. DOI: 10.1186/s42492-019-0016-7. URL: https://vciba.springeropen.com/articles/10.1186/s42492-019-0016-7.

Faragó, Tomáš et al. (May 2022). " Tofu : a fast, versatile and user-friendly image processing toolkit for computed tomography ". In: *Journal of Synchrotron Radiation* 29.3, pp. 916–927. ISSN: 1600-5775. DOI: 10.1107/s160057752200282x. URL: https://scripts.iucr.org/cgi-bin/paper?S160057752200282X.

Fitzgerald, Richard (July 2000). "Phase-Sensitive X-Ray Imaging". In: *Physics Today* 53.7, pp. 23–26. ISSN: 0031-9228. DOI: 10.1063/1.1292471.

Fuchs, Patrick, Thorben Kröger, and Christoph S. Garbe (Sept. 2021). "Defect detection in CT scans of cast aluminum parts: A machine vision perspective". In: *Neurocomputing* 453, pp. 85–96. ISSN: 18728286. DOI: 10.1016/j.neucom.2021.04.094. URL: https://linkinghub.elsevier.com/retrieve/pii/S0925231221006524.

Fundamentals, DOE (1993). *Nuclear physics and reactor theory*. Tech. rep. Technical Report.

Geirhos, Robert et al. (Nov. 2018). "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness". In.

Getzin, Matthew et al. (2018). "Increased separability of K-edge nanoparticles by photon-counting detectors for spectral micro-CT". In: *Journal of X-ray science and technology* 26.5, pp. 707–726.

Gonzalez, Rafael C. and Richard E. Woods (2008). *Digital Image Processing*. Prentice Hall. ISBN: 9780131687288 013168728X 9780135052679 013505267.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. The MIT Press, p. 800. ISBN: 9780262035613.

Gu, Shuhang and Radu Timofte (2019). "A brief review of image denoising algorithms and beyond". In: *Inpainting and Denoising Challenges*, pp. 1–21.

Gupta, Vipul et al. (Apr. 2022). "SwapMix: Diagnosing and Regularizing the Over-Reliance on Visual Context in Visual Question Answering". In.

Haghighi, Fatemeh et al. (Feb. 2021). "Transferable Visual Words: Exploiting the Semantics of Anatomical Patterns for Self-supervised Learning". In: *IEEE Transactions on Medical Imaging*. ISSN: 1558254X. DOI: 10.1109/TMI.2021.3060634. URL: http://arxiv.org/abs/2102.10680.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2009). *The Elements of Statistical Learning*. Springer Series in Statistics. New York, NY: Springer New York. ISBN: 978-0-387-84857-0. DOI: 10.1007/978-0-387-84858-7. URL: http://link.springer.com/10.1007/978-0-387-84858-7.

He, Kaiming et al. (Dec. 2016a). "Deep residual learning for image recognition". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2016-Decem, pp. 770–778. ISSN: 10636919. DOI: 10.1109/CVPR.2016.90. URL: http://arxiv.org/abs/1512.03385.

— (2016b). "Deep residual learning for image recognition". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol. 2016-Decem. DOI: 10.1109/CVPR.2016.90.

He, Kaiming et al. (Mar. 2020). "Mask R-CNN". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.2, pp. 386–397. ISSN: 19393539. DOI: 10.1109/TPAMI.2018.2844175. URL: http://arxiv.org/abs/1703.06870.

Helgason, Sigurdur (1965). "The Radon transform on Euclidean spaces, compact two-point homogeneous spaces and Grassmann manifolds". In: *Acta Mathematica* 113, pp. 153–180.

Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean (Mar. 2015). "Distilling the Knowledge in a Neural Network". In: URL: http://arxiv.org/abs/1503.02531.

Hsu, Cheng-Chun et al. (2019). "Weakly Supervised Instance Segmentation using the Bounding Box Tightness Prior". In: *Advances in Neural Information Processing Systems 32*. Ed. by H Wallach et al. Curran Associates, Inc., pp. 6586–6597.

URL: http://papers.nips.cc/paper/8885-weakly-supervised-instance-segmentation-using-the-bounding-box-tightness-prior.pdf.

Huisken, Jan et al. (Aug. 2004). "Optical Sectioning Deep Inside Live Embryos by Selective Plane Illumination Microscopy". In: *Science* 305.5686, pp. 1007–1009. ISSN: 0036-8075. DOI: 10.1126/science.1100035.

Ilesanmi, Ademola E. and Taiwo O. Ilesanmi (Oct. 2021). "Methods for image denoising using convolutional neural network: a review". In: *Complex & Intelligent Systems* 7.5, pp. 2179–2198. ISSN: 2199-4536. DOI: 10.1007/s40747-021-00428-4. URL: https://link.springer.com/10.1007/s40747-021-00428-4.

Jing, Longlong and Yingli Tian (Feb. 2021). "Self-Supervised Visual Feature Learning with Deep Neural Networks: A Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.11, pp. 4037–4058. ISSN: 19393539. DOI: 10.1109/TPAMI.2020.2992393. URL: http://arxiv.org/abs/1902.06162.

Jørgensen, Jakob S et al. (2021). "Core Imaging Library – Part I: a versatile Python framework for tomographic imaging". In: *Philosophical Transactions of the Royal Society A* 379.2204, p. 20200192.

Jørgensen, JS et al. (2019). "Neutron TOF imaging phantom data to quantify hyperspectral reconstruction algorithms". In: *STFC ISIS Neutron and MuonSource*.

Kak, Avinash C and Malcolm Slaney (2001). *Principles of computerized tomographic imaging*. SIAM.

Kamp, Thomas van de et al. (Aug. 2018). "Parasitoid biology preserved in mineralized fossils". In: *Nature Communications* 9.1, p. 3325. ISSN: 20411723. DOI: 10.1038/s41467-018-05654-y. URL: https://www.nature.com/articles/s41467-018-05654-y.

Kamp, Thomas van de (2011). "Functional morphology of the weevil genus Trigonopterus (Coleoptera: Curculionidae)." PhD thesis. Heinrich Heine University Düsseldorf, p. 155.

Keklikoglou, Kleoniki et al. (Sept. 2021). "Micro-ct for biological and biomedical studies: A comparison of imaging techniques". In: *Journal of Imaging* 7.9, p. 172. ISSN: 2313433X. DOI: 10.3390/jimaging7090172. URL: https://www.mdpi.com/2313-433X/7/9/172.

Khan, Asifullah et al. (Jan. 2020). "A survey of the recent architectures of deep convolutional neural networks". In: *Artificial Intelligence Review* 53.8, pp. 5455–5516. ISSN: 15737462. DOI: 10.1007/s10462-020-09825-6. URL: http://arxiv.org/abs/1901.06032http://dx.doi.org/10.1007/s10462-020-09825-6.

Kim, Tae Soo et al. (Sept. 2022). "Did You Get What You Paid For? Rethinking Annotation Cost of Deep Learning Based Computer Aided Detection in Chest Radiographs". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 13433 LNCS, pp. 261–270. ISSN: 16113349. DOI: 10.1007/978-3-031-16437-8{_}25. URL: http://arxiv.org/abs/2209.15314.

Kingma, Diederik P. and Jimmy Lei Ba (2015). "Adam: A method for stochastic optimization". In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Kinoshita, Masato et al. (June 2009). *Medaka*. Wiley. ISBN: 9780813808710. DOI: 10.1002/9780813818849.

Kirillov, Alexander et al. (Apr. 2023). "Segment Anything". In: URL: http://arxiv.org/abs/2304.02643.

Kiryati, Nahum and Yuval Landau (Aug. 2021). "Dataset growth in medical image analysis research". In: *Journal of Imaging* 7.8, p. 155. ISSN: 2313433X. DOI: 10.3390/jimaging7080155. URL: https://www.mdpi.com/2313-433X/7/8/155.

Kockelmann, Winfried et al. (2018). "Time-of-Flight Neutron Imaging on IMAT@ISIS: A New User Facility for Materials Science". In: *Journal of Imaging* 4.3. ISSN: 2313-433X. DOI: 10.3390/jimaging4030047. URL: https://www.mdpi.com/2313-433X/4/3/47.

Kockelmann, Winifred et al. (2007). "Energy-selective neutron transmission imaging at a pulsed source". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 578.2, pp. 421–434.

Krull, Alexander, Tim Oliver Buchholz, and Florian Jug (Nov. 2019). "Noise2void-Learning denoising from single noisy images". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2019-June, pp. 2124–2132. ISSN: 10636919. DOI: 10.1109/CVPR.2019.00223. URL: http://arxiv.org/abs/1811.10980.

Kwee, Thomas C. and Robert M. Kwee (Nov. 2020). "Chest CT in COVID-19: What the Radiologist Needs to Know". In: *RadioGraphics* 40.7, pp. 1848–1865. ISSN: 0271-5333. DOI: 10.1148/rg.2020200159.

Larsson, Gustav, Michael Maire, and Gregory Shakhnarovich (Mar. 2017). "Colorization as a proxy task for visual understanding". In: *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017* 2017-Janua, pp. 840–849. DOI: 10.1109/CVPR.2017.96. URL: http://arxiv.org/abs/1703.04044.

Lecun, Y. et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324. ISSN: 00189219. DOI: 10.1109/5.726791. URL: http://ieeexplore.ieee.org/document/726791/.

Lehtinen, Jaakko et al. (Mar. 2018). "Noise2Noise: Learning image restoration without clean data". In: *35th International Conference on Machine Learning, ICML 2018* 7, pp. 4620–4631. URL: http://arxiv.org/abs/1803.04189.

Lohse, Leon M. et al. (May 2020). "A phase-retrieval toolbox for X-ray holography and tomography". In: *Journal of Synchrotron Radiation* 27.3, pp. 852–859. ISSN: 16005775. DOI: 10.1107/S1600577520002398. URL: https://scripts.iucr.org/cgi-bin/paper?S1600577520002398.

Lösel, Philipp D. et al. (Dec. 2020). "Introducing Biomedisa as an open-source online platform for biomedical image segmentation". In: *Nature Communications* 11.1, p. 5577. ISSN: 20411723. DOI: 10.1038/s41467-020-19303-w. URL: http://www.nature.com/articles/s41467-020-19303-w.

Luca, Andreea Roxana et al. (2022). "Impact of quality, type and volume of data used by deep learning models in the analysis of medical images". In: *Informatics in Medicine Unlocked* 29, p. 100911. ISSN: 23529148. DOI: 10.1016/j.imu.2022.100911. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352914822000612.

Ma, Jun et al. (Mar. 2021). "Toward data-efficient learning: A benchmark for COVID-19 CT lung and infection segmentation". In: *Medical Physics* 48.3, pp. 1197–1210. ISSN: 00942405. DOI: 10.1002/mp.14676. URL: https://onlinelibrary.wiley.com/doi/10.1002/mp.14676.

MacMahon, Heber et al. (Nov. 2005). "Guidelines for management of small pulmonary nodules detected on CT scans: A statement from the Fleischner Society". In: *Radiology* 237.2, pp. 395–400. ISSN: 00338419. DOI: 10.1148/radiol.2372041887. URL: http://pubs.rsna.org/doi/10.1148/radiol.2372041887.

McCulloch, Warren S. and Walter Pitts (Dec. 1943). "A logical calculus of the ideas immanent in nervous activity". In: *The Bulletin of Mathematical Biophysics* 5.4, pp. 115–133. ISSN: 0007-4985. DOI: 10.1007/BF02478259. URL: http://link.springer.com/10.1007/BF02478259.

Moosmann, Julian et al. (May 2013). "X-ray phase-contrast in vivo microtomography probes new aspects of Xenopus gastrulation". In: *Nature* 497.7449, pp. 374–377. ISSN: 0028-0836. DOI: 10.1038/nature12116. URL: http://www.nature.com/articles/nature12116.

Murphy, Kevin P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press, p. 1104. ISBN: 9780262018029.

Narla, Akhila et al. (Oct. 2018). "Automated Classification of Skin Lesions: From Pixels to Practice". In: *Journal of Investigative Dermatology* 138.10, pp. 2108–2110. ISSN: 0022202X. DOI: 10.1016/j.jid.2018.06.175.

Newell, Alejandro and Jia Deng (Mar. 2020). "How useful is self-supervised pretraining for visual tasks?" In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 7343–7352. ISSN: 10636919. DOI: 10.1109/CVPR42600.2020.00737. URL: http://arxiv.org/abs/2003.14323.

Noroozi, Mehdi and Paolo Favaro (Mar. 2016). "Unsupervised learning of visual representations by solving jigsaw puzzles". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9910 LNCS, pp. 69–84. ISSN: 16113349. DOI: 10.1007/978-3-319-46466-4{_}5. URL: http://arxiv.org/abs/1603.09246.

OpenAI (Mar. 2023). "GPT-4 Technical Report". In: URL: http://arxiv.org/abs/2303.08774.

Oquab, Maxime et al. (Apr. 2023). "DINOv2: Learning Robust Visual Features without Supervision". In: URL: http://arxiv.org/abs/2304.07193.

Paganin, D. et al. (Apr. 2002). "Simultaneous phase and amplitude extraction from a single defocused image of a homogeneous object". In: *Journal of Microscopy* 206.1, pp. 33–40. ISSN: 0022-2720. DOI: 10.1046/j.1365-2818.2002.01010.x.

Paganin, D et al. (2004). "Quantitative phase-amplitude microscopy. III. The effects of noise". In: *Journal of microscopy* 214.1, pp. 51–61.

Papkov, Mikhail et al. (Nov. 2021). "Noise2Stack: Improving Image Restoration by Learning from Volumetric Data". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 12964 LNCS, pp. 99–108. ISSN: 16113349. DOI: 10.1007/978-3-030-88552-6{_}10. URL: http://arxiv.org/abs/2011.05105.

Papoutsellis, Evangelos et al. (Aug. 2021). "Core Imaging Library - Part II: multichannel reconstruction for dynamic and spectral tomography". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 379.2204, p. 20200193. ISSN: 1364-503X. DOI: 10.1098/rsta.2020.0193.

Pavel Iakubovskii (2019). *Segmentation Models Pytorch*. URL: https://github.com/qubvel/segmentation_models.pytorch.

Pedregosa, Fabian et al. (Jan. 2011). "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830.

Power, Stephen P et al. (2016). "Computed tomography and patient risk: Facts, perceptions and uncertainties". In: *World Journal of Radiology* 8.12, p. 902. ISSN: 1949-8470. DOI: 10.4329/wjr.v8.i12.902. URL: http://www.wjgnet.com/1949-8470/full/v8/i12/902.htm.

Prakash, Mangal et al. (Apr. 2022). "Interpretable Unsupervised Diversity Denoising and Artefact Removal". In: *International Conference on Learning Representations*. URL: http://arxiv.org/abs/2104.01374.

Protonotarios, Nicholas E. et al. (2021). "Piecewise Polynomial Inversion of the Radon Transform in Three Space Dimensions via Plane Integration and Applications in Positron Emission Tomography". In: *Springer Optimization and Its Applications*.

Vol. 173, pp. 381–396. DOI: 10.1007/978-3-030-72563-1{_}17. URL: https://link.springer.com/10.1007/978-3-030-72563-1_17.

Punnoose, Jacob et al. (2016). "spektr 3.0—A computational tool for x-ray spectrum modeling and analysis". In: *Medical physics* 43.8Part1, pp. 4711–4717.

Radon, Johann (Dec. 1986). "on the Determination of Functions From Their Integral Values Along Certain Manifolds." In: *IEEE Transactions on Medical Imaging* MI-5.4, pp. 170–176. ISSN: 02780062. DOI: 10.1109/tmi.1986.4307775. URL: http://ieeexplore.ieee.org/document/4307775/.

Rajaraman, Sivaramakrishnan et al. (May 2021). "Chest X-ray Bone Suppression for Improving Classification of Tuberculosis-Consistent Findings". In: *Diagnostics* 11.5, p. 840. ISSN: 2075-4418. DOI: 10.3390/diagnostics11050840.

Ranzato, Marc'Aurelio et al. (June 2007). "Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition". In: *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1–8. ISBN: 1-4244-1179-3. DOI: 10.1109/CVPR.2007.383157. URL: http://ieeexplore.ieee.org/document/4270182/.

Ren, Pengzhen et al. (Aug. 2022). "A Survey of Deep Active Learning". In: *ACM Computing Surveys* 54.9. ISSN: 15577341. DOI: 10.1145/3472291. URL: http://arxiv.org/abs/2009.00236.

Rodríguez, Paul (2013). "Total variation regularization algorithms for images corrupted with different noise models: a review". In: *Journal of Electrical and Computer Engineering* 2013.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (May 2015). "U-net: Convolutional networks for biomedical image segmentation". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9351, pp. 234–241. ISSN: 16113349. DOI: 10.1007/978-3-319-24574-4{_}28. URL: http://arxiv.org/abs/1505.04597.

Rudin, Leonid I, Stanley Osher, and Emad Fatemi (1992). "Nonlinear total variation based noise removal algorithms". In: *Physica D: nonlinear phenomena* 60.1-4, pp. 259–268.

Russakovsky, Olga et al. (Sept. 2015). "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision* 115.3, pp. 211–252. ISSN: 15731405. DOI: 10.1007/s11263-015-0816-y. URL: http://arxiv.org/abs/1409.0575.

Santisteban, J. R. et al. (June 2001). "Time-of-flight neutron transmission diffraction". In: *Journal of Applied Crystallography* 34.3, pp. 289–297. ISSN: 0021-8898. DOI: 10.1107/S0021889801003260.

Santisteban, Javier R et al. (2002). "Engineering applications of Bragg-edge neutron transmission". In: *Applied Physics A* 74.1, s1433–s1436.

Schofield, R. et al. (May 2020). "Image reconstruction: Part 1 – understanding filtered back projection, noise and image acquisition". In: *Journal of Cardiovascular Computed Tomography* 14.3, pp. 219–225. ISSN: 1876861X. DOI: 10.1016/j.jcct.2019.04.008. URL: https://linkinghub.elsevier.com/retrieve/pii/S1934592519300607.

Sculley, D (2009). "Large Scale Learning to Rank". In: *NIPS 2009 Workshop on Advances in Ranking*, pp. 1–6.

Selvaraju, Ramprasaath R. et al. (Oct. 2016). "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization". In: DOI: 10.1007/s11263-019-01228-7.

Shanthanagouda, Admane H. et al. (Feb. 2014). "Japanese Medaka: A Non-Mammalian Vertebrate Model for Studying Sex and Age-Related Bone Metabolism In Vivo". In: *PLoS ONE* 9.2, e88165. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0088165.

Shen, Yunhang et al. (2019). "Cyclic guidance for weakly supervised joint detection and segmentation". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol. 2019-June. DOI: 10.1109/CVPR.2019.00079.

Sidky, Emil Y, Chien-Min Kao, and Xiaochuan Pan (2006). "Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT". In: *Journal of X-ray Science and Technology* 14.2, pp. 119–139.

Simonyan, Karen and Andrew Zisserman (Sept. 2015). "Very deep convolutional networks for large-scale image recognition". In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. URL: http://arxiv.org/abs/1409.1556.

Sombke, Andy et al. (June 2015). "Potential and limitations of X-Ray micro-computed tomography in arthropod neuroanatomy: A methodological and comparative survey". In: *Journal of Comparative Neurology* 523.8, pp. 1281–1295. ISSN: 0021-9967. DOI: 10.1002/cne.23741.

Spiecker, Rebecca et al. (2023a). "Bragg magnifier based dose-efficient in vivo X-ray imaging at micrometer resolution". In: *submitted for publication*.

Spiecker, Rebecca et al. (Dec. 2023b). "Dose-efficient in vivo X-ray phase contrast imaging at micrometer resolution by Bragg magnifiers". In: *Optica* 10.12, p. 1633. ISSN: 2334-2536. DOI: 10.1364/OPTICA.500978.

Spitzer, Hannah et al. (June 2018). "Improving cytoarchitectonic segmentation of human brain areas with self-supervised siamese networks". In: *Lecture Notes in Computer Science* 11072 LNCS, pp. 663–671. ISSN: 16113349. DOI: 10.1007/978-3-030-00931-1{_}76. URL: http://arxiv.org/abs/1806.05104.

Strobl, M et al. (2009). "Advances in neutron radiography and tomography". In: *Journal of Physics D: Applied Physics* 42.24, p. 243001.

Tao, Siwei et al. (Mar. 2021). "Principles of Different X-ray Phase-Contrast Imaging: A Review". In: *Applied Sciences* 11.7, p. 2971. ISSN: 2076-3417. DOI: 10.3390/app11072971. URL: https://www.mdpi.com/2076-3417/11/7/2971.

Tarvainen, Antti and Harri Valpola (Mar. 2017). "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results". In: *Advances in Neural Information Processing Systems* 2017-Decem, pp. 1196–1205. ISSN: 10495258. URL: http://arxiv.org/abs/1703.01780.

Tremsin, Anton S et al. (2012). "High Resolution Photon Counting With MCP-Timepix Quad Parallel Readout Operating at > 1 kHz Frame Rates". In: *IEEE transactions on nuclear science* 60.2, pp. 578–585.

Tuszynski, J (2006). *2022 PhotonAttenuation–Software for modeling of photons passing through different materials*. URL: https://de.mathworks.com/matlabcentral/fileexchange/12092-photonattenuation.

Valverde, Juan Miguel et al. (Apr. 2021). "Transfer Learning in Magnetic Resonance Brain Imaging: A Systematic Review". In: *Journal of Imaging* 7.4, p. 66. ISSN: 2313-433X. DOI: 10.3390/jimaging7040066.

Van De Kamp, Thomas et al. (July 2011). "A biological screw in a beetle's leg". In: *Science* 333.6038, p. 52. ISSN: 00368075. DOI: 10.1126/science.1204245. URL: https://www.science.org/doi/10.1126/science.1204245.

Vásárhelyi, L. et al. (Dec. 2020). "Microcomputed tomography–based characterization of advanced materials: a review". In: *Materials Today Advances* 8, p. 100084.

ISSN: 25900498. DOI: 10.1016/j.mtadv.2020.100084. URL: https://linkinghub.elsevier.com/retrieve/pii/S259004982030031X.

Vincent, Pascal et al. (2008). "Extracting and composing robust features with denoising autoencoders". In: *Proceedings of the 25th International Conference on Machine Learning*. New York, New York, USA: ACM Press, pp. 1096–1103. ISBN: 9781605582054. DOI: 10.1145/1390156.1390294. URL: http://portal.acm.org/citation.cfm?doid=1390156.1390294.

Wang, Yu et al. (2021). "Improving Self-supervised Learning with Automated Unsupervised Outlier Arbitration". In: *Advances in Neural Information Processing Systems*. URL: http://arxiv.org/abs/2112.08132.

Wang, Z. et al. (Apr. 2004). "Image Quality Assessment: From Error Visibility to Structural Similarity". In: *IEEE Transactions on Image Processing* 13.4, pp. 600–612. ISSN: 1057-7149. DOI: 10.1109/TIP.2003.819861.

Warr, Ryan et al. (2021). "Enhanced hyperspectral tomography for bioimaging by spatiospectral reconstruction". In: *Scientific reports* 11.1, pp. 1–13.

Weinhardt, Venera et al. (Dec. 2018). "Quantitative morphometric analysis of adult teleost fish by X-ray computed tomography". In: *Scientific Reports* 8.1, p. 16531. ISSN: 2045-2322. DOI: 10.1038/s41598-018-34848-z. URL: http://www.nature.com/articles/s41598-018-34848-z.

Weng, Lilian (2021). *Contrastive Representation Learning*. URL: https://lilianweng.github.io/lil-log/2021/05/31/contrastive-representation-learning.html.

Wolf, Ivo et al. (May 2004). "The medical imaging interaction toolkit (MITK): a toolkit facilitating the creation of interactive software by extending VTK and ITK". In: *Medical Imaging 2004: Visualization, Image-Guided Procedures, and Display*. Ed. by Robert L. Galloway Jr. Vol. 5367, p. 16. DOI: 10.1117/12.535112. URL: http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.535112.

Xie, Qizhe et al. (Nov. 2020). "Self-training with noisy student improves imagenet classification". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695. ISSN: 10636919. DOI: 10.1109/CVPR42600.2020.01070. URL: http://arxiv.org/abs/1911.04252.

Yuan, Li et al. (Sept. 2020). "Revisiting Knowledge Distillation via Label Smoothing Regularization". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3902–3910. ISSN: 10636919. DOI: 10.1109/CVPR42600.2020.00396. URL: http://arxiv.org/abs/1909.11723.

Zettler, Nico and Andre Mastmeyer (July 2021). "Comparison of 2D vs. 3D U-Net Organ Segmentation in abdominal 3D CT images". In: *Computer Science Research Notes* 3101, pp. 41–50. ISSN: 24644625. DOI: 10.24132/CSRN.2021.3101.5. URL: http://arxiv.org/abs/2107.04062.

Zhang, Richard, Phillip Isola, and Alexei A. Efros (Mar. 2016). "Colorful image colorization". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9907 LNCS, pp. 649–666. ISSN: 16113349. DOI: 10.1007/978-3-319-46487-9{_}40. URL: http://arxiv.org/abs/1603.08511.

Zharov, Yaroslav, Tilo Baumbach, and Vincent Heuveline (Mar. 2023). "Optimizing the Procedure of CT Segmentation Labeling". In: URL: http://arxiv.org/abs/2303.14089.

Zharov, Yaroslav et al. (Mar. 2022). "Using the Order of Tomographic Slices as a Prior for Neural Networks Pre-Training". In: URL: http://arxiv.org/abs/2203.09372.

Zharov, Yaroslav et al. (July 2023). "Shot noise reduction in radiographic and tomographic multi-channel imaging with self-supervised deep learning". In: *Optics Express* 31.16, p. 26226. ISSN: 1094-4087. DOI: 10.1364/OE.492221.

Zhuang, Fuzhen et al. (Nov. 2021). "A Comprehensive Survey on Transfer Learning". In: *Proceedings of the IEEE* 109.1, pp. 43–76. ISSN: 15582256. DOI: 10.1109/JPROC.2020.3004555. URL: http://arxiv.org/abs/1911.02685.

Zong, Bo et al. (2018). "Deep autoencoding Gaussian mixture model for unsupervised anomaly detection". In: *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, pp. 1–19.

Zoph, Barret et al. (June 2020). "Rethinking pre-training and self-training". In: *Advances in Neural Information Processing Systems* 33. ISSN: 10495258. URL: http://arxiv.org/abs/2006.06882.

Zou, Yang et al. (2018b). "Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-training". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. ISBN: 9783030012182. DOI: 10.1007/978-3-030-01219-9{_}18.

— (Oct. 2018a). "Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-training". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11207 LNCS, pp. 297–313. ISSN: 16113349. DOI: 10.1007/978-3-030-01219-9{_}18. URL: http://arxiv.org/abs/1810.07911.