DISSERTATION
submitted
to the
Combined Faculty for the Natural
Sciences and Mathematics
of
Heidelberg University, Germany
for the degree of
Doctor of Natural Sciences

Put forward by
Dipl. Alexander Kirillov
Born in Moscow, Russia
Oral examination:

Exploring Aspects of Image Segmentation: Diversity, Global Reasoning, and Panoptic Formulation

Advisor: Prof. Dr. Carsten Rother

Acknowledgments

First and foremost, I would like to thank my main supervisor Carsten Rother for encouraging me to think about global computer vision questions. His enthusiasm and support have helped me to approach new ambitious research projects without fear and hesitation. Above all, he created an exciting atmosphere in our lab in Dresden and then in Heidelberg.

I am very much obliged to Dmitry Vetrov for introducing me to the fascinating world of graphical models and for showing me how much fun scientific discussions can actually be. Also I would like to thank Bogdan Savchynskyy for his supervision. I could not ask for a better senior colleague and friend. He has fostered my progress and helped me on each and every step. Beyond research collaboration, it was he who introduced me to bouldering and uphill running which I appreciate greatly.

I would like to thank all members of the CVLD and VLL labs for the openness and great atmosphere. Dima for the remarkably deep discussions of research work we had during our billiard games. Frank, Eric and Alex for making me feel at home in Germany. Hassan, Omid and Siva for the wonderful overnight discussions and support during deadline sprints. Sid, Stefan and Lisa for sharing excitement about graphical models. Jakob for Vision pub organization. All of them for their proof reading efforts and help with rehearsals.

I am also very much indebted to my best friend Michael who I started my research path with at Moscow State University. His insights and critical remarks have helped me a lot.

Last but not least I would like to thank my family for their unconditional support despite the distances that have often separated us. During my PhD I met my wife Maria. I am extremely grateful for her limitless love, encouragement, trust, and care. For tolerating my deadline sprints (especially as the CVPR deadline usually during the same week as my wife's birthday) and for reminding me that there is life beyond research work.

Abstract

Image segmentation is the task of partitioning an image into *meaningful* regions. It is a fundamental part of the visual scene understanding problem with many real-world applications, such as photo-editing, robotics, navigation, autonomous driving and bio-imaging. It has been extensively studied for several decades and has transformed into a set of problems which define meaningfulness of regions differently. The set includes two high-level tasks: semantic segmentation (each region assigned with a semantic label) and instance segmentation (each region representing object instance). Due to their practical importance, both tasks attract a lot of research attention. In this work we explore several aspects of these tasks and propose novel approaches and new paradigms.

While most research efforts are directed at developing models that produce a single best segmentation, we consider the task of producing multiple diverse solutions given a single input image. This allows to hedge against the intrinsic ambiguity of segmentation task. We propose a new global model with multiple solutions for a trained segmentation model. This new model generalizes previously proposed approaches for the task. We present several approximate and exact inference techniques that suit a wide spectrum of possible applications and demonstrate superior performance comparing to previous methods.

Then, we present a new bottom-up paradigm for the instance segmentation task. The new scheme is substantially different from the previous approaches that produce each instance independently. Our approach named InstanceCut reasons globally about the optimal partitioning of an image into instances based on local clues. We use two types of local pixel-level clues extracted by efficient fully convolutional networks: (i) an instance-agnostic semantic segmentation and (ii) instance boundaries. Despite the conceptual simplicity of our approach, it demonstrates promising performance.

Finally, we put forward a novel Panoptic Segmentation task. It unifies semantic and instance segmentation tasks. The proposed task requires generating a coherent scene segmentation that is rich and complete, an important step towards real-world vision systems. While early work in computer vision addressed related image/scene parsing tasks, these are not currently popular, possibly due to lack of appropriate metrics or associated recognition challenges. To address this, we first offer a novel panoptic quality metric that captures performance for all classes (stuff and things) in an interpretable and unified manner. Using this metric, we perform a rigorous study of both human and machine performance for panoptic segmentation on three existing datasets, revealing interesting insights about the task. The aim of our work is to revive the interest of the community in a more unified view of image segmentation.

Zusammenfassung

In der Bildsegmentierung besteht die Aufgabe darin, ein Bild in inhaltlich sinnvolle Regionen einzuteilen. Damit ist sie für die Bildverarbeitung von hoher Bedeutung und findet in zahlreichen Bereichen, beispielsweise bei der Fotoaufbereitung, in der Robotik, in der Navigation, beim autonomen Fahren sowie in der Biologie, Anwendung. Im Laufe der seit einigen Jahrzehnten stattfindenden Forschung zur Bildsegmentierung haben sich verschiedene Problemformulierungen herauskristallisiert, die sich darin unterscheiden, wie Regionen inhaltlich definiert sind. Zwei dieser Aufgaben sind semantische Segmentierung (jede Region erhält eine semantische Bezeichnung) und Instanzsegmentierung (jede Region stellt eine Objektinstanz dar). Aufgrund ihrer praktischen Bedeutung haben beide Problemstellungen in der Forschung bereits viel Aufmerksamkeit erhalten. In der vorliegenden Arbeit stellen wir einige ihrer Aspekte vor und schlagen neue Herangehensweisen und Ansätze vor.

Im Gegensatz zum weit verbreiteten Forschungsansatz, Modelle zu entwickeln, die eine einzige bestmögliche Segmentierung liefern, betrachten wir die Aufgabe, zu einem gegebenen Eingangsbild mehrere verschiedenartige Lösungen zu generieren. Dadurch ist es möglich, die immanente Mehrdeutigkeit des Segmentierungsproblems zu berücksichtigen. Wir führen ein neues globales Modell ein, welches für ein trainiertes Segmentierungsmodell mehrere Lösungen liefert. Es verallgemeinert bereits bestehende Ansätze für das genannte Problem. Wir stellen mehrere näherungsweise und exakte Inferenztechniken vor, die für eine große Spanne möglicher Anwendungen genutzt werden können, und zeigen, dass sie bisherigen Methoden überlegen sind.

Außerdem stellen wir einen neuen Bottom-Up-Ansatz für die Instanzsegmentierung vor. Dieser unterscheidet sich wesentlich von bisherigen Herangehensweisen, welche jede Instanz einzeln erzeugen. Unser InstanceCut genannter Ansatz sucht anhand lokaler Merkmale global nach einer optimalen Partitionierung des Bildes in Instanzen. Dafür nutzen wir zwei Typen lokaler pixelbasierter Merkmale, die mit Hilfe von Fully Convolutional Networks extrahiert werden: (i) eine Instanz-unabhängige semantische Segmentierung und (ii) Instanzübergänge. Obwohl diese Herangehensweise konzeptionell einfach ist, liefert sie vielversprechende Ergebnisse.

Abschließend führen wir das neuartige panoptische Segmentierungsproblem ein. Es vereint semantische und Instanzsegmentierung. Für das vorgeschlagene Problem ist es erforderlich, eine schlüssige Szenensegmentierung zu generieren, die vollständig und reichhaltig ist – ein wichtiger Schritt in Richtung praktisch anwendbarer Bildverarbeitungssysteme. Obwohl frühere Arbeiten auf dem Gebiet der Bildverarbeitung bereits ähnliche Bildanalyseaufgaben betrachtet haben, sind diese momentan kaum verbreitet, was möglicherweise am Fehlen geeigneter Metriken oder damit verbundener Bilderkennungs-Wettbewerbe liegt. Um dem zu begegnen, schlagen wir zunächst

ein neuartiges panoptisches Qualitätsmaß vor, welches auf einheitliche und nachvollziehbare Weise die Performance für alle Klassen (Bereiche sowie Objekte) bewertet. Diese Metrik ermöglicht uns einen fundierten Vergleich menschlicher und maschineller Kompetenz in der panoptischen Segmentierung auf drei bestehenden Datensätzen, wodurch interessante Erkenntnisse über dieses Problem offengelegt werden. Ziel dieser Arbeit ist es, das Interesse der Forschungsgemeinde an einer vereinheitlichten Sicht auf die Bildsegmentierung wiederzubeleben.

Contents

Acknowledgments **5**

Abstract **7**

Zusammenfassung **10**

1 Introduction **13**
1.1 Image Segmentation Challenges 16
1.1.1 Multiple Diverse Solutions 16
1.1.2 Global Reasoning for Instance Segmentation 20
1.1.3 Segmentation for Scene Understanding Applications 22
1.2 Contribution 23
1.3 List of Published Research Papers 24
1.4 Outline of The Thesis 26

2 Multiple Diverse Solutions Inference **27**
2.1 Introduction 27
2.2 Related Work 28
2.3 General Multiple Diverse Solutions Problem 29
2.3.1 Formulation 30
2.3.2 Connection to DivMBest [Bat+12] 30
2.3.3 Connection to DPP [KT10] 31
2.4 Formal Problem Definition 32
2.4.1 Energy minimization 32
2.4.2 Diversity Measure 33
2.4.3 General Diversity Optimization Problem 34
2.5 Optimization Techniques 34
2.5.1 Greedy Approach: DivMBest [Bat+12] 35
2.5.2 Clique Encoding 36
2.5.3 Ordering Based Approach 39
2.5.4 Parametric based Approach 44
2.6 Experimental Evaluation 49
2.6.1 Datasets 50
2.6.2 Clique Encoding 51
2.6.3 Ordering Based 51
2.6.4 Parametric Based 53
2.7 Conclusion 55

3 Bottom-Up Approach for Instance Segmentation **57**
3.1 Introduction . . . 57
3.2 Related Work . . . 59
3.3 InstanceCut . . . 60
3.3.1 Overview of the proposed framework . . . 60
3.3.2 Semantic Segmentation . . . 62
3.3.3 Instance-Aware Edge Detection . . . 62
3.3.4 Image Partition . . . 65
3.4 Experiments . . . 68
3.5 Discussion . . . 70

4 Panoptic Segmentation **73**
4.1 Introduction . . . 73
4.2 Related Work . . . 75
4.3 Panoptic Segmentation Format . . . 77
4.4 Panoptic Segmentation Metric . . . 78
4.4.1 Segment Matching . . . 78
4.4.2 Panoptic Quality (PQ) Computation . . . 79
4.4.3 Comparison to Existing Metrics . . . 80
4.5 Panoptic Segmentation Datasets . . . 81
4.6 Human Performance Study . . . 82
4.7 Machine Performance Baselines . . . 85
4.8 Future of Panoptic Segmentation . . . 89

5 Discussion **91**
5.1 Limitations and Future Work . . . 91
5.1.1 Multiple Diverse Solutions . . . 92
5.1.2 Bottom-Up Instance Segmentation Framework . . . 93
5.1.3 Segmentation for Scene Understanding Applications . . . 94

Bibliography **97**

Chapter 1

Introduction

Humans perceive the visual world via a complex system that starts from our eyes. Photoreceptor cells on the retina of the human eye convert light that hits the retina into neural impulses. Among these cells cone cells are responsible for a sharp color visual signal. Densely packed on the central part of the retina three types of cone cells convert red, green, and blue components of light into neural impulses. The human visual perception system then interprets this dense map of neural impulses to be able to act inside the environment. Although there are still a lot of open research questions regarding exact mechanisms of human visual perception, it is clear that we are able to extract rich scene information from the point-wise color map representing visual input.

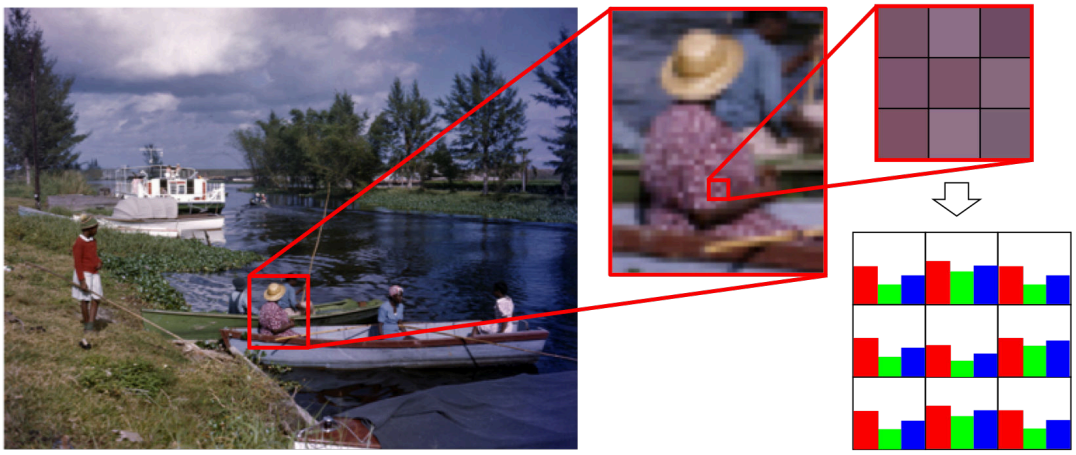


Figure 1.1: **RGB pixel encoding of an image.** Computers store the image as a grid of pixels. In each pixel three values correspond to red, green, and blue components of the pixel color.

Computer representation of an image somewhat resembles the neural impulses map created by cone photoreceptor cells. An example is shown in Fig. 1.1. For a computer an image is a grid of pixels where each pixel has its color. Pixel colors can be encoded differently. As presented in this example, the RGB scheme decomposes each color into three components: red, green, and blue; this is a direct approximation of the three types of cone cells. In the same way as neural impulses from cone cells are the basic input of the human visual system, this pixel representation is the basic input of computer vision systems.

One of the goals of computer vision is to build automatic systems that are able to imitate human perception by extracting high-level scene information from an image or video. Grouping the elements of visual input is an example of this high-level information. Almost 100 years ago, studying the human visual perception system, Wertheimer [Wer23] explored the ways in which we group some visual elements and perceive them as a whole. He described several principals of this grouping such as proximity, similarity, and common behavior. The computer vision counterpart of this perceptual grouping task is called image segmentation.

According to David Marr [Mar82] the notion of image segmentation is a "division of the image into regions that are meaningful either for the purpose at hand or for their correspondence to physical objects or their parts". This notion captures the idea that image segmentation is not a single well-defined task. Diverse applications constitute different definitions of "meaningfulness":

- Super-pixel image segmentation (Fig. 1.2) aims to split the image into regions that are visually consistent with respect to local clues such as brightness, color, and textures. These regions may be treated as intermediate image representation (super-pixels) used by high-level scene understanding tasks.

Figure 1.2: Super-pixel segmentation output [Ach+12] for the image from ADE20k [Zho+17].

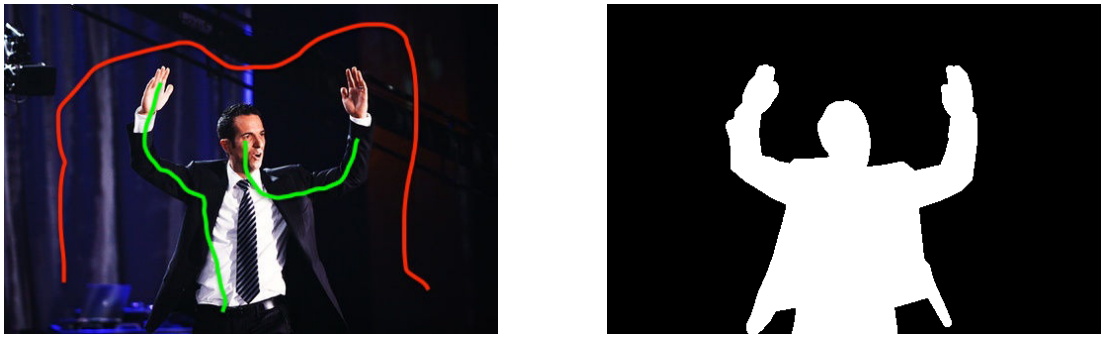


Figure 1.3: Foreground/background segmentation example with additional user supervision. Image from VOC2009 [Eve+15]. The user provides clues for foreground/background separation using brush strokes.

- Foreground/background segmentation (Fig. 1.3) aims to extract the image's region of interest (foreground) based on some additional input. A practical example

of this task is photo editing where a user wants to change the background of an image.

- Semantic Segmentation (Fig. 1.4) aims to group pixels according to a set of semantic labels like "road", "buildings", "cars", etc. This task provides information about the whole scene that can be used for autonomous driving, robotics, and medical applications. Semantic segmentation can be equivalently formulated as a task of assigning semantic labels to all pixels in the image. We will use this formulation further in the text.

Figure 1.4: Semantic segmentation examples for the image from ADE20k [Zho+17]. Different colors represent different semantic labels. Among others the set of semantic labels contains “dining table”, “chair”, “wall”, “tile-floor”.

These are just a few examples of well-known image segmentation tasks. Multiple new challenges like instance-aware semantic segmentation [Lin+14] (segment each object instance separately) and segmentation of 3D bio-images [Men+14] (3D scans of human tissues) have become very popular driven by practical needs. In general, image segmentation can be seen as a first step of a complex computer vision system converting a grid of pixels into meaningful regions that are then used to solve the task at hand including navigation[Cor+16], photo editing[RKB04], or biomedical applications [RFB15].

A large number of methods was developed to solve image segmentation problems. They have been of interest to the research community for almost half a century. With the increasing amount of available computational power and training data, multiple paradigms were explored during this period of time including classical clustering methods [HS85], variational formulations [BZ87], normalized cuts [SM00], Conditional Random Fields (CRFs) [WJ08] and more recently approaches based on Convolutional Neural Networks (CNNs) [LSD15]. Details of the methods differ significantly depending on the segmentation task at hand.

Today, the two most common paradigms for semantic-based image segmentation are CRFs and CNNs. CRFs allow to impose additional constraints on the resulting segmentation based on expert knowledge about the task. These constraints force solutions to comply with some known structure of the desired segmentation. Incorporating this additional knowledge enables to generalize using less training data. CNNs are mechanisms to learn powerful feature representations directly from data. Multiple

benchmarks show the superiority of CNN based approaches for the task where large sets of training data are available.

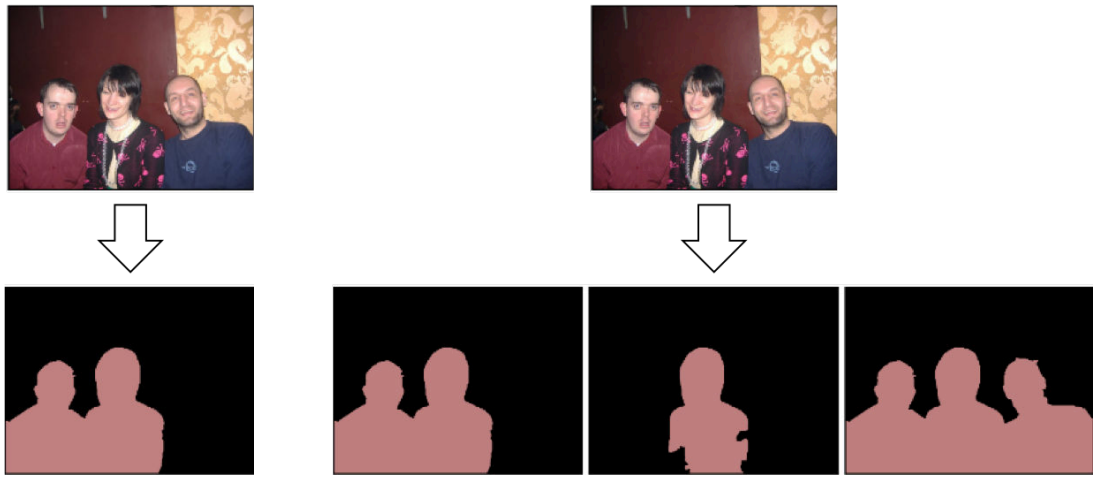
In this work we explore solutions that use both the CRF and CNN paradigms together. Chapters 2 and 3 of this thesis provide formal definitions of these frameworks applied to the task of image segmentation.

1.1 Image Segmentation Challenges

Image segmentation has been explored for almost half a century. However, it is still an active area of research. While for some sub-tasks like super-pixel image segmentation modern techniques achieve very high performance [LJK17], for other tasks a decent performance level requires massive sets of annotated data that are not always available or are very expensive to obtain. Moreover, with performance saturation for the standard tasks, more challenging segmentation tasks, like instance-aware semantic segmentation [Lin+14; Cor+16], have appeared. For these new tasks there is ample room for future improvements. In this work we focus on several aspects of image segmentation tasks that in our opinion require new breakthroughs. In what follows we briefly introduce these challenges and summarize our contribution.

1.1.1 Multiple Diverse Solutions

Most current semantic image segmentation techniques operate according to the following paradigm: given an image they produce a function that assigns a score to every possible segmentation of the image. The final output is either the exact or approximate optimum of this function. Following pioneering work in this direction [Bat+12], we argue that there are cases in which finding multiple solutions (that are diverse) for the same input image is desirable (see Fig. 1.5). We present several such cases below.



(a) (b)

Figure 1.5: Semantic image segmentation examples: (a) single best segmentation according to a trained model, (b) multiple segmentations for the same input image.

Data ambiguity. One of the reasons to produce multiple solutions is the intrinsic ambiguity of segmentation tasks. For instance, boundaries between objects can be fuzzy or simply unclear (an example is shown in Fig. 1.6 top row). Moreover, sometimes it is not possible to assign the right semantic label to a segment without additional context (see Fig. 1.6 bottom row). Creators of several modern semantic segmentation datasets [Cor+16; Zho+17; CUF18] report the level of inconsistency between different annotators producing ground truth for the same image. For instance, in [Zho+17] on average 16% of pixels get different semantic labels when the same image is annotated two times independently.

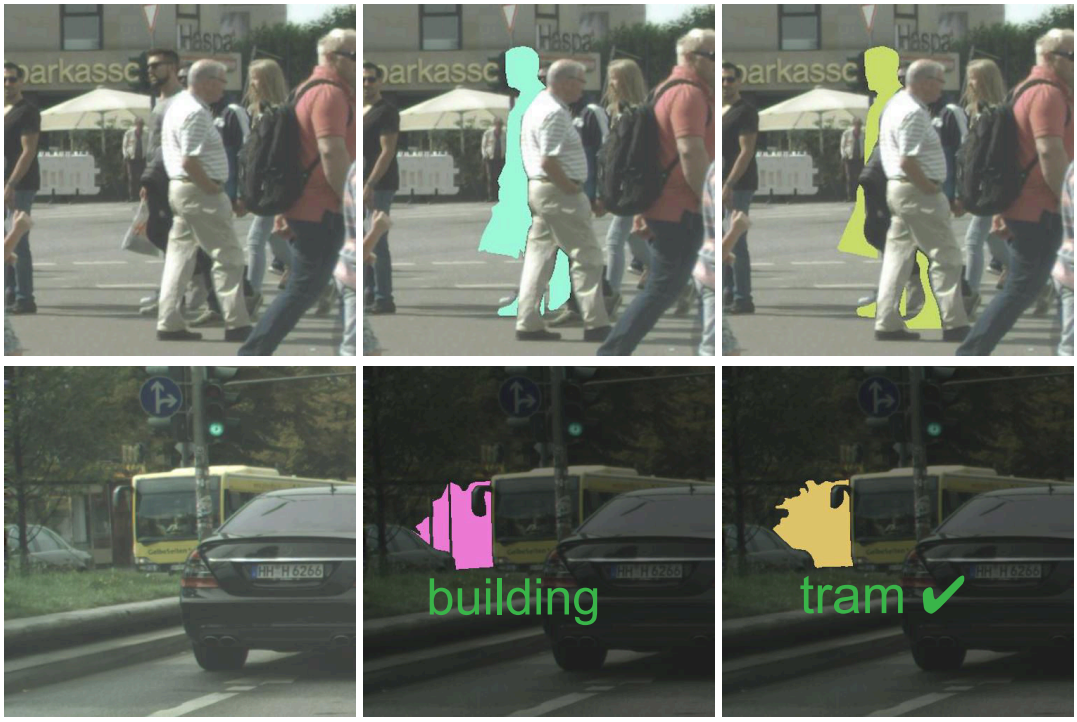


Figure 1.6: Semantic segmentation ambiguity. (Cityscapes [Cor+16]) Images are zoomed and cropped. Top row: the segmentation of the person is genuinely ambiguous. Bottom row: the scene is extremely difficult, tram is the correct class for the segment.

(a) input with user scribbles (b) possible solution (c) another possible solution

Figure 1.7: Interactive segmentation ambiguity. (Pacal VOC [Eve+15].) Based on provided user supervision, it is not possible to determine which of the two possible answers is correct .

Interactive foreground/background segmentation used extensively in photo-editing

tools [RKB04] is another example of a highly ambiguous task. In this scenario, a user provides supervision for the segmentation in the form of two types of brush strokes that mark areas belonging to foreground or background respectively. Fig. 1.7 (a) illustrates an image and user supervision for foreground (green strokes) and background (blue strokes). For this input the right answer cannot be determined. It is unclear whether the user wants to segment out a single car of the train or the whole train.

Currently the majority of segmentation frameworks does not take this ambiguity into account [LSD15; Che+17a; YK16]. These methods treat inconsistencies as noise in ground truth annotations. In contrast, methods that produce multiple solutions are able to hedge against the data ambiguity.

Poor models / lack of data. Most segmentation models are trained in a discriminative fashion so that solutions with the best score/probability correspond to the most accurate results. However, as noted in [Sze+08; Bat+12] during test time a solution with a worse score may be more accurate than the one with the best score. This may be explained by approximation error (model capacity is not sufficient to learn all nuances) or estimation error (training data is limited and does not allow to fit the true data distribution). As it was shown in [Bat+12] and later in our works as well [Kir+15a; Kir+15b], other solutions that have good but not the best scores according to the trained model may be more accurate (Fig. 1.8 illustrates this situation).

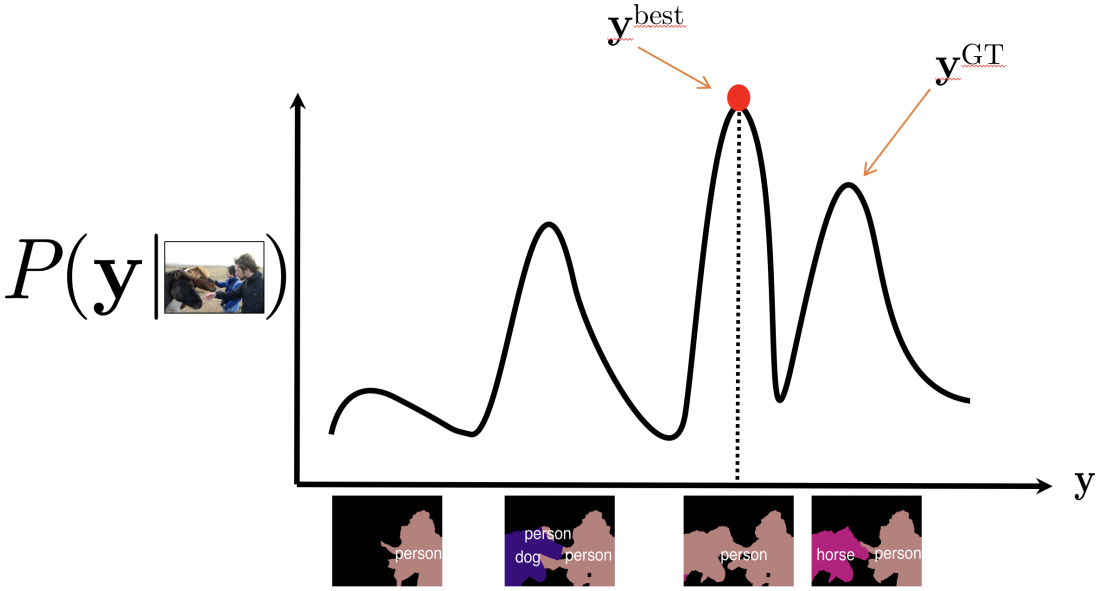
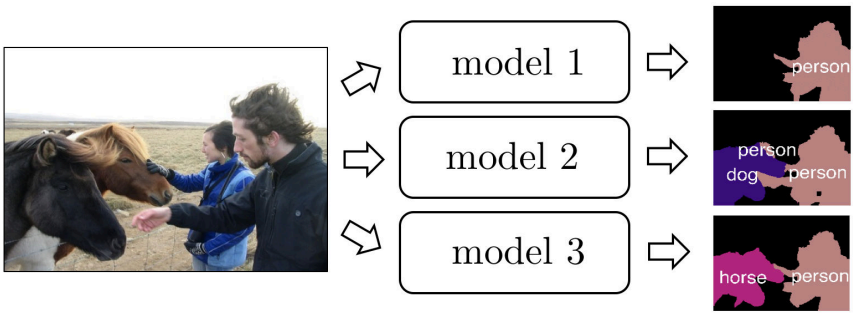


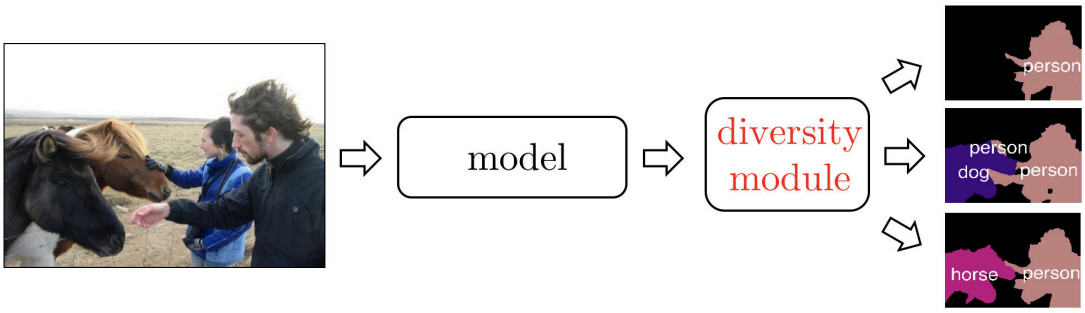
Figure 1.8: Given an image a segmentation model produces a function that assigns a score to each possible solution. The solution with the best score may actually be less accurate than another solution with a worse score as shown here.

Existing methods. Previous works propose two main ways to produce multiple diverse solutions given a single input: training-stage diversity and inference-stage diversity, see Fig. 1.9. The first option proposes to simultaneously train multiple models each producing a single solution [GRBK12; Guz+14; Lee+16]. The second option is

to infer multiple solutions from a model trained to produce a single solution [Bat+12; Kir+15a; Che+13]. Both cases have their own pros and cons. While training several models requires more computational resources, it gives additional flexibility, *i.e.* the way solutions differ may be controlled directly. Inferring multiple solutions from a single model is less flexible, but requires less computational power and less space to store the model. Moreover, in this case there is no need to have access to the training procedure that may be unavailable. We discuss advantages and disadvantages of the existing methods in more detail in Chapter 2.



(a) training-stage diversity



(b) inference-stage diversity

Figure 1.9: Two main approaches to produce multiple diverse solutions for a single input: (a) training-stage diversity and (b) inference-stage diversity.

Applications. Multiple diverse solutions can be used directly as a final output in certain applications that assume interaction with users [Bat+12] or as a part of a bigger pipeline where these solutions are used by the next stages of the pipeline. For instance, multiple solutions can be applied to speed up cutting-plane optimization [GRKB13] or estimate uncertainty [RB12]. A more general example is a pipeline where the first stage produces multiple solutions, and then the next stage selects the best one using additional knowledge [LCK18; YBS13]. A basic representation of such a system is depicted in Fig. 1.10.

The development of new holistic methods that are able to produce multiple diverse solutions serves two important goals: (1) to make computer vision systems more robust given limited training data and (2) to incorporate knowledge of intrinsic ambiguity of visual perception tasks directly into vision systems.

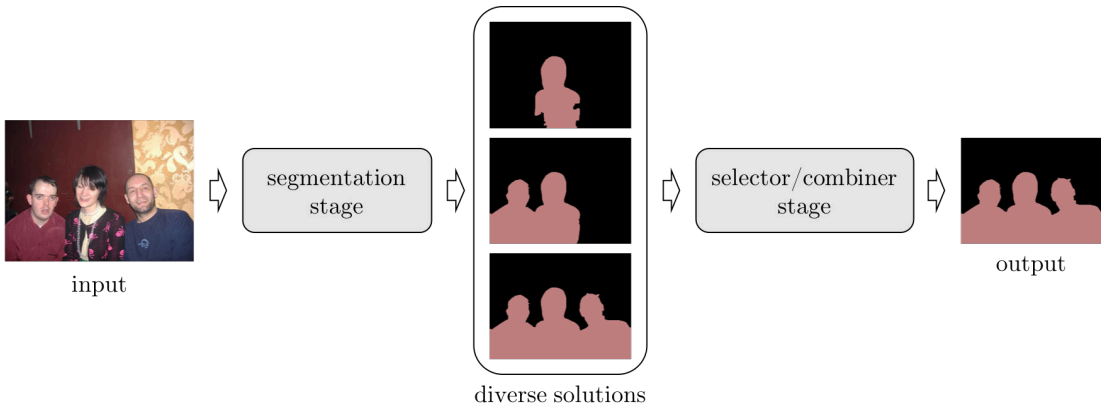


Figure 1.10: General usage of multiple solutions in a bigger pipeline. The first stage produces multiple solutions given a single input and then the second stage selects a single solution or combines these solutions into one.

1.1.2 Global Reasoning for Instance Segmentation

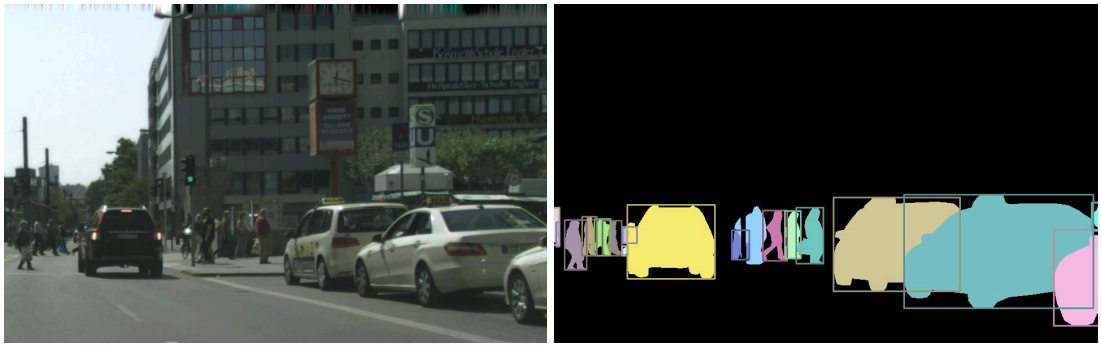


Figure 1.11: Instance segmentation example. Pixels that belong to the same instance of the “car” or “pedestrian” semantic category share a color.

Instance-aware semantic segmentation or simply instance segmentation is a relatively new member of the image segmentation tasks family. The task can be seen as an evolution of the well-known bounding box detection task that aims to delineate object instances by bounding boxes. The goal is to identify individual objects in the scene with pixel-level accuracy (Fig. 1.11). It was recently popularized by several large-scale datasets [Lin+14; Cor+16] that provide pixel-level masks for each instance of semantic categories like “car”, “person”, etc. Instance segmentation is defined only for categories that have the notion of instances, *i.e.* “things” categories. Unlike semantic segmentation that will group all pixels that correspond to "person" in one segment, instance segmentation groups pixels that correspond to different persons separately. Segmentation of instances can then be used to analyze object behavior and possible actions.

Most of the current state-of-the-art instance segmentation approaches leverage the successful bounding-boxes detection methods. They either generate bounding boxes first and then use a binary segmentation method to delineate instances inside each bounding box separately [Har+14; He+17], or generate proposal instance masks first

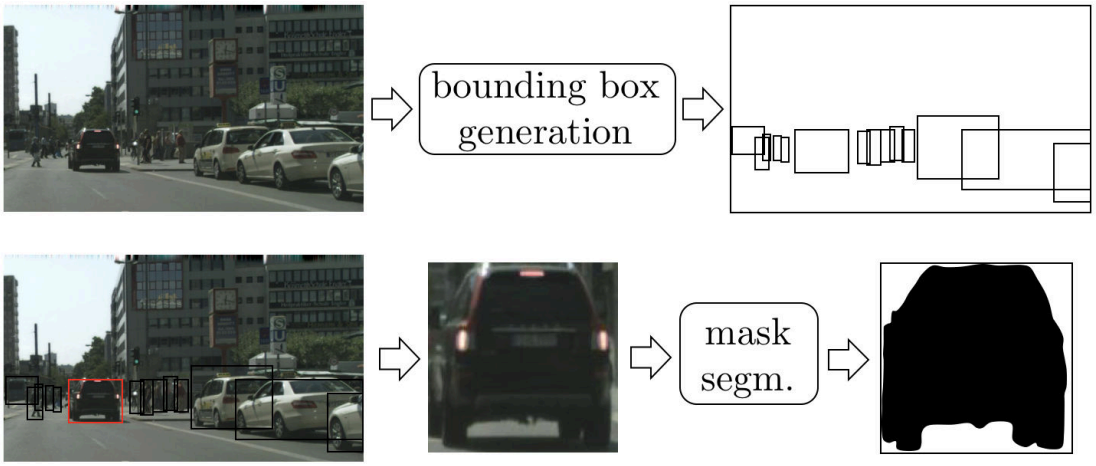


Figure 1.12: Scheme of the top-down instance segmentation approach. First, bounding boxes are generated, then for each bounding box independently a segmentation network performs binary segmentation. The instances predicted from all bounding boxes form the final prediction.

and then filter them using a classification method [Car+12]. This type of method is called top-down approach, since it first detects objects globally and then refines each object independently. The general scheme of a top-down approach is depicted in Fig. 1.12. Top-down instance segmentation methods inherit the recognition power from bounding-box detection methods. Thanks to that, these methods are often able to find very small and distant objects. While quite powerful, top-down approaches are not always able to utilize global context or object relations to segment hard cases. To overcome these issues, global reasoning techniques that rearrange and filter obtained proposals with respect to co-occurrence were recently proposed [Hu+18]. Despite being limited by the set of obtained proposals, these methods have demonstrated promising results.

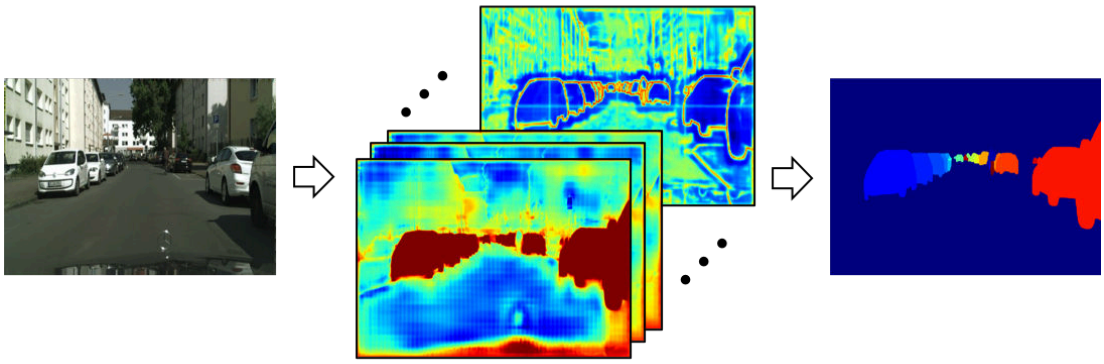


Figure 1.13: Scheme of the bottom-up instance segmentation approach. First, local clues are extracted on a pixel-level, then single global reasoning produces an instance segmentation for the whole image.

One possible alternative to the top-down paradigm is a bottom-up scheme. Instead

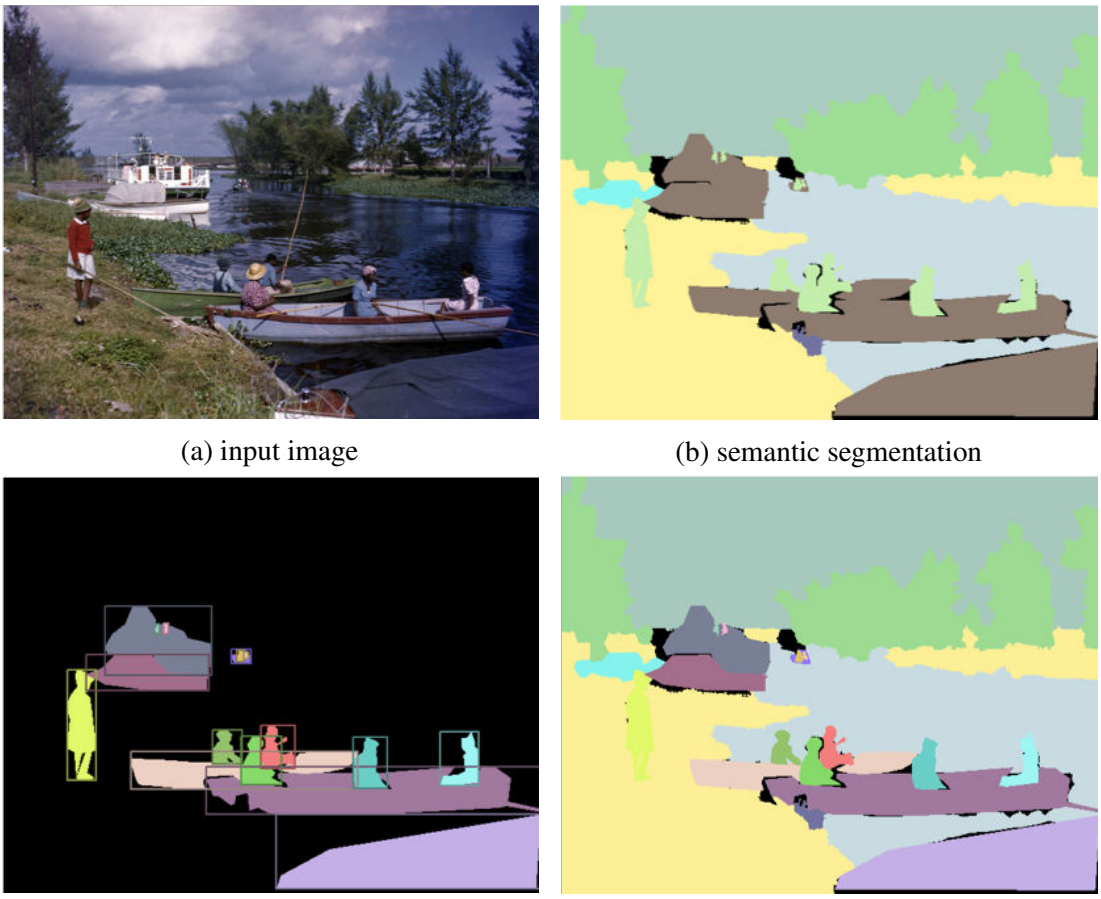
of detecting objects independently, it first extracts some local clues on a per-pixel basis and then these clues are used to infer all instances via one global reasoning procedure (see Fig. 1.13). Global inference in this paradigm provides the ability to make coherent prediction and to make a combined decision instead of many individual predictions without additional context about surrounding decisions. Moreover, the approach based on this scheme can directly use semantic segmentation methods to produce required pixel-level clues. Any improvement of quality in semantic segmentation techniques will help the bottom-up method as well.

Currently the general scheme of the bottom-up approach is mainly popular for problems other than instance segmentation. For example, great performance was demonstrated by a bottom-up approach for a key-point human pose estimation task [Cao+17]. The main obstacle in the adoption of this paradigm for instance segmentation is the lack of general global inference techniques for the task. Existing greedy approaches [Uhr+16] are not able to utilize the full potential of the scheme. Exploration of novel bottom-up approaches for instance segmentation and their combination with top-down approaches is a fundamental step forward towards robust and practically applicable recognition systems that successfully utilize context and real-world knowledge.

1.1.3 Segmentation for Scene Understanding Applications

Nowadays instance segmentation and semantic segmentation are the two main high-level segmentation tasks. Multiple modern segmentation datasets [Cor+16; Zho+17; Neu+17] have both instance and semantic ground truth annotations with two separate challenges for instance and semantic segmentation respectively. Both tasks extract viable information from an image that is used in computer vision systems. Providing semantic labels for each pixel on the image, semantic segmentation helps to infer important details of the image including scene type and geometric properties. On the other hand object masks inferred by an instance segmentation method are needed to analyze the behavior of instances and their relations. Multiple real-world applications need complimentary information about the input scene that these two segmentation tasks provide. For instance, in an autonomous driving scenario the semantic segmentation output is needed to identify drivable areas. At the same time, it needs instance-level information about surrounding cars and pedestrians for avoiding collisions and navigating.

Several earlier works proposed methods that simultaneously produce semantic and instance segmentation [YFU12; TL13; TNL14; Sun+14] (see illustration of simultaneous segmentation in Fig. 1.14). However, despite its significant practical relevance the joint task has not become popular. In our point of view, the main reason is the absence of a quality metric that evaluates performance of such a joint method in a uniform way. For the most part researchers have explored semantic and instance segmentation separately. Given significant interest from industry and availability of large scale datasets with both semantic and instance segmentation annotations, the development of a new performance metric for the challenge will in our opinion attract research attention to the combined task.



(a) input image (b) semantic segmentation

(c) instance segmentation (d) combined output

Figure 1.14: For a given image (a), we show ground truth for: (b) semantic segmentation (per-pixel class labels), (c) instance segmentation (per-object mask and class label), and (d) combined instance and semantic segmentation ground truth.

1.2 Contribution

In this work we focus on several aspects of image segmentation described in the previous sections. In what follows we shortly summarize the main contributions of this thesis. The detailed technical contributions are presented in Chapters 2 to 4.

- We propose a new problem formulation for the inference of multiple diverse solutions from a single trained model as well as the algorithms for its solution:
  - Our formulation generalizes most of the previously proposed approaches to the diversity problem. This includes, but is not limited to the determinant point processes [KT10] and the DivMBest method [Bat+12]. The former is a special case of our formulation, whereas the latter can be seen as a greedy algorithm for solving the diversity problem in our formulation.
  - We propose several exact and approximate algorithms to solve the diversity problem in our generalized formulation. These algorithms vary from more general and slow to more specific and fast ones. The former address a broader class of problems, whereas the latter require certain properties by

the underlying model and the diversity measure to be fulfilled. Notably, we show that our algorithms provide solutions of higher quality, since they address the diversity problem in our new rigorous formulation.

  - An interesting theoretical result, which we obtain here, is the close relation of our diversity problem formulation and the class of parametric submodular minimization problems [FI03; Bac13]. The latter are known also as parametric max-flow [GGT89; Hoc08] in a special case. We show that under certain technical conditions, multiple diverse solutions can be obtained as a result of submodular parametric minimization. This yields an extremely efficient diversity algorithm and shows a tight relation between these two seemingly unrelated areas.

- We introduce a novel bottom-up paradigm for instance segmentation. First, local clues are extracted from an image, then a new global reasoning technique infers all instances simultaneously. Local pixel-level information is extracted by two classifiers: a semantic segmentation network and a boundary detection network. The first provides a score for each pixel and each semantic label and the second one computes the likelihood of a boundary between any two neighboring pixels. The global reasoning inference for the instance segmentation is formulated as a graph partitioning problem, where graph nodes stand for (super-)pixels of an input image, edges connect neighboring (super-)pixels of the image and the node and edge weights are determined by the above classifiers. In spite of the simplicity of the formulation, our approach shows competitive results and performs particularly well on rare object classes.

- We propose a Panoptic Segmentation problem formulation that combines the semantic and instance segmentations into a single consistent task. The new task aims to generate segmentation that is richer than output of each task individually and is consistent at the same time. As a part of the task, we introduce the novel Panoptic Quality performance measure. This new quality measure is simple and intuitive. It treats categories with and without instance notion in a uniform manner. Moreover, it allows to measure human performance for panoptic segmentation task directly. We perform a rigorous experimental evaluation of this new measure and task on several popular segmentation datasets to show its practical relevance.

1.3 List of Published Research Papers

The remaining chapters of the thesis are based on the following research papers.

1. **Inferring M-Best Diverse Labelings in a Single One**
   Alexander Kirillov, Bogdan Savchynskyy, Dmitrij Schlesinger, Dmitry Vetrov, Carsten Rother
   IEEE International Conference on Computer Vision (ICCV) 2015

2. **M-Best-Diverse Labelings for Submodular Energies and Beyond**
   Alexander Kirillov, Dmitrij Schlesinger, Dmitry Vetrov, Carsten Rother, Bogdan Savchynskyy
   Advances in Neural Information Processing Systems (NIPS) 2015

3. **Joint M-Best-Diverse Labelings as a Parametric Submodular Minimization**
   Alexander Kirillov, Alexander Shekhovtsov, Carsten Rother, Bogdan Savchynskyy
   Advances in Neural Information Processing Systems (NIPS) 2016

4. **InstanceCut: from Edges to Instances with MultiCut**
   Alexander Kirillov, Evgeny Levinkov, Bjoern Andres, Bogdan Savchynskyy, Carsten Rother
   IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017

5. **Panoptic Segmentation**
   Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, Piotr Dollár
   arXiv preprint arXiv:1801.00868

We also contributed to the following papers associated with image segmentation. However, we will not discuss them in the thesis.

6. **Conditional Random Fields Meet Deep Neural Networks for Semantic Segmentation: Combining Probabilistic Graphical Models with Deep Learning for Structured Prediction**
   Anurag Arnab, Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Mans Larsson, Alexander Kirillov, Bogdan Savchynskyy, Carsten Rother, Fredrik Kahl, Philip HS Torr
   IEEE Signal Processing Magazine (SPM) 2018

7. **Analyzing Modular CNN Architectures for Joint Depth Prediction and Semantic Segmentation**
   Omid Hosseini Jafari, Oliver Groth, Alexander Kirillov, Michael Ying Yang, Carsten Rother
   IEEE International Conference on Robotics and Automation (ICRA) 2017

8. **Joint Training of Generic CNN-CRF Models with Stochastic Optimization**
   Alexander Kirillov, Dmytro Schlesinger, Shuai Zheng, Bogdan Savchynskyy, Philip HS Torr, Carsten Rother
   Asian Conference on Computer Vision (ACCV) 2016

During the work on this thesis, we have also contributed to the following papers that are on topics other than image segmentation.

9. **A Comparative Study of Local Search Algorithms for Correlation Clustering**
   Evgeny Levinkov, Alexander Kirillov, Bjoern Andres
   German Conference on Pattern Recognition (GCPR) 2017

10. **Global hypothesis generation for 6D object pose estimation**
    Frank Michel, Alexander Kirillov, Eric Brachmann, Alexander Krull, Stefan Gumhold, Bogdan Savchynskyy, Carsten Rother
    IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017

11. **Joint Graph Decomposition & Node Labeling: Problem, Algorithms, Applications**
    Evgeny Levinkov, Jonas Uhrig, Siyu Tang, Mohamed Omran, Eldar Insafutdinov, Alexander Kirillov, Carsten Rother, Thomas Brox, Bernt Schiele, Bjoern Andres
    IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017

12. **Deep Part-Based Generative Shape Model with Latent Variables**
    Alexander Kirillov, Mikhail Gavrikov, Ekaterina Lobacheva, Anton Osokin, Dmitry Vetrov
    British Machine Vision Conference (BMVC) 2016

1.4 Outline of The Thesis

The remaining part of this work is structured as follows: Chapter 2 introduces our approach to producing multiple diverse solutions from a single trained model. Here we describe new optimization techniques for different types of models. In Chapter 3 we present a novel bottom-up instance segmentation approach. We demonstrate its competitive performance on a challenging autonomous driving dataset, Cityscapes [Cor+16]. Chapter 4 is devoted to the novel Panoptic Segmentation task. We explore the properties of the task on three major segmentation datasets. We discuss contributions of this thesis and outline some limitations and future directions in Chapter 5.

Chapter 2

Multiple Diverse Solutions Inference

2.1 Introduction

A number of computer vision and machine learning tasks can be seen as a task of selecting best suited output $\boldsymbol{y}$ from a predefined set $\mathcal{Y}$ for an input. Computer vision examples of such tasks are image classification and image segmentation. In this thesis we focus on segmentation tasks, however, described techniques can be applied for other applications as well. A trained model for image segmentation problem usually assigns a score or probability for each possible segmentation output given an input. One can always represent the score assignment as a function $E(\boldsymbol{y}) : \mathcal{Y} \to \mathbb{R}$; then, the best output according to the model can be found by solving the following optimization problem:

$$\arg\min_{\boldsymbol{y} \in \mathcal{Y}} E(\boldsymbol{y}) \,. \tag{2.1}$$

We assume here that the best output according to the trained model has the smallest score. Using common notation, we will call the function $E(\boldsymbol{y})$ *energy function* and score corresponding to $\boldsymbol{y}$ – *energy* of $\boldsymbol{y}$. The optimization problem (2.1) is also called *Maximum A Posteriori (MAP) inference*. If a trained model returns a probability p for each output, then the energy can be obtained as $-\log(p)$. Note, that almost any trained model can be represented in the form of (2.1). For instance, both Conditional Random Fields (CRFs) [WJ08] and Convolutional Neural Networks [LSD15] can be written as (2.1).

Image segmentation research is mostly focused on the ways of training the best possible model, i.e., obtaining $E(\boldsymbol{y})$ such that the solutiuon of (2.1) has the best perfromace accoridng to a target metric. During the last decade, novel deep learning approaches have drastically improved results for image segmentation tasks [LSD15; YK16; Che+17a]. Using large annotated datasets, these techniques demonstrate remarkable boost of performance. The effort of training a model to get the best possible energy function focuses on obtaining single best solution for a task.

Several works explore an orthogonal direction of obtaining several good solutions for a given input instead of a single one. This setup hedges against errors caused by intrinsic ambiguity of a real-world task or limited availability of the training data. Note, that classic formulation (2.1) cannot possibly solve an issue like ambiguity, since it

must return a single result for any given input. There are two main approaches to obtain multiple solutions for a given input: training several models to return different results [GRBK12; Guz+14; Lee+16] or infering several solutions from a single model trained to infer a single solution only [Bat+12; Kir+15a; Che+13]. While former is more flexible, it's less computationally efficient than latter. In our work we focus on efficiency and, thus, on the latter option.

Natural question is how the multiple solutions for a single input may be used in practice. Firstly, the most obvious application is an interactive scenario where a user can select the most suitable option [Bat+12]. Secondly, multiple solutions are used to estimate uncertainty [RB12] or speed-up training [GRKB13]. Lastly, multiple solutions can be used as a step in the middle of a pipeline, where they will be filtered, re-weighted or combined using additional information [YBS13; PTB14; LCK18].

Our work generalizes over existing research in the area of producing multiple diverse solutions for a single input. We provide a road map that will hopefully guide future researchers showing what optimization options and what guarantees they have depending on their specific problems. With our work we aim to facilitate the usage of multiple solutions for the existing applications and inspire ideas for new research directions. We summarize contributions of this chapter as follows:

- We introduce a novel general problem formulation of obtaining several good solutions from a single trained model. Given a trained model in the form of energy function $E(\boldsymbol{y})$, instead of optimizing for a single solution as in (2.1), we form a new optimization problem to infer M solutions simultaneously. We show that our model generalizes previously developed techniques that produce multiple solutions [Bat+12; KT10].

- We present approximate global optimization technique for the new task that is applicable to a broad range of problems and demonstrates superior performance comparing with previous approaches.

- For submodular original energies $E(\boldsymbol{y})$ we offer new optimization techniques that produce multiple diverse solutions solving the new optimization problem exactly and faster that previous approximate approaches [Bat+12].

2.2 Related Work

M-Best solutions. The problem of obtaining M solutions with the best energies according to an energy function $E(\boldsymbol{y})$ has been of interest to our research community for a long time. Back in 1972, a procedure of computing *M-best solutions* or *M-Best MAP inference* problem was proposed in [Law72]. Later, more efficient techniques were developed. They worked with special subclasses of energy functions: tree-shaped graphical models [SH02, Ch. 8], junction-trees [Nil98] and general graphical models [YW04; FG09; Bat12]. M-best solutions inference methods are well-suited for a problem with a small set of possible solutions $\mathcal{Y}$; however, for a pixel-labelling problem like semantic segmentation, where $\mathcal{Y}$ has exponential size, M-best solutions are often nearly identical and, hence, have no practical use.

Sampling approaches. Energy of a solution can be seen as negative logarithm of unnormalized probability. Using this representation of a probability distribution over possible solutions, different sampling schemes are applicable to obtain M solutions that are highly probable according to the energy function. Early work in this direction introduced local Gibbs sampling scheme [GG84]. Later, schemes with much better mixing time were proposed [PZ11; TZ02]. These techniques can approximate uncertainty of the energy function by sampling multiple solutions. Yet, they don't explicitly force solutions to be sufficiently different from each other; therefore, they often require a lot of solutions to be sampled in order to cover different modes of the underlying distribution. Modern *Perturb-and-map* sampling method [PY11] is much more efficient. It requires multiple MAP-inference problems to be solved exactly and, therefore, is applicable only if the exact inference can be performed very fast.

Diversity solutions. Structured Determinantal Point Processes (SDPP) [KT10] defines probability distribution over sets of solutions so that sets with diverse low-energy solutions have high probability. In SDPP, efficient sampling is only possible if underlying model has a tree-structure. Several methods of obtaining M *best modes* [Che+13] are applicable to the same narrow class of models. In our work we explore methods applicable to a broader range of models.

The closest to our work is DivMBest approach [Bat+12; PJB14]. The work proposes to obtain M diverse solutions sequentially by solving sequence of problems like (2.1) with additional terms that forces new solution to be far away from previously obtained solutions according to some diversity measures. DivMBest is applicable to general graphical models and efficient optimization techniques for several diversity measures were introduced in [Bat+12; PJB14]. Obtaining solutions one by one, the method has a greedy nature. In our work we show that more integrated approach outperforms the greedy scheme.

Training of M *independent* models to produce diverse solutions was proposed in [GRBK12; Guz+14]. M solutions are obtained by solving (2.1) for each trained model. Explicit control over training procedures for the models gives more freedom and ability to satisfy some specific properties. On the other hand, M models slow down both training and inference stages and also increase memory consumption. In our work, we assume *a single fixed* model supporting reasonable MAP-solutions. Our approach doesn't require an access to training procedure.

2.3 General Multiple Diverse Solutions Problem

Several different approaches were developed for the problem of obtaining M diverse solutions from a single energy function $E(\boldsymbol{y})$. These methods have various pros and cons, and their efficiency depends on the particular application. Natural question is *how one can select the best-suited approach for a specific task?* In our work we propose generalized view on the problem. We formulate single optimization problem and show that existing methods are special cases of the problem. Further we discuss existing optimization schemes, propose new techniques and explore their limitations. We aim to ease for a final user the problem of selecting the best approach given specific needs of the application in hand.

2.3.1 Formulation

We start by identifying several simple desiderata for diverse solutions we want to obtain from a single model represented by an energy function $E(\boldsymbol{y})$:

- Each solution has a good (low) energy according to the model;
- We wish the solutions to be diverse.

We define novel optimization problem that contains two terms to fulfill the desiderata. First term is the sum of energies of M solutions $\sum_{m=1}^{M} E(\boldsymbol{y}^m)$. By minimizing this term we aim to get M solutions with the lowest possible energies. Second term is diversity measure $\Delta^M(\boldsymbol{y}^1, \dots, \boldsymbol{y}^M)$ that takes a large value if solutions $\boldsymbol{y}^1, \dots, \boldsymbol{y}^M$ are diverse, in a certain sense, and a small value otherwise. Both terms together form the following optimization problem:

$$\underset{(\boldsymbol{y}^1,\dots,\boldsymbol{y}^M)\in\mathcal{Y}^M}{\arg\min} \sum_{m=1}^{M} E(\boldsymbol{y}^m) - \lambda\Delta^M(\boldsymbol{y}^1, \dots, \boldsymbol{y}^M)\,, \tag{2.2}$$

where scalar $\lambda > 0$ determines a trade-off between these two terms. We call (2.2) *General Multiple Diverse Solutions Problem*. The optimization problem (2.2) encode described desiderata in the most straightforward way. The sum of the energy functions forces solutions to have the lowest possible energies. At the same time the second term forces the solutions to be diverse in a certain sense that is defined by function $\Delta^M(\boldsymbol{y}^1, \dots, \boldsymbol{y}^M)$. One of the common examples of diversity measure is the sum of Hamming distances between all solutions. In the next sections we show that the new optimization problem is, in fact, a generalization over previously proposed methods for diverse solutions: DivMBest [Bat+12] and DPP [KT10].

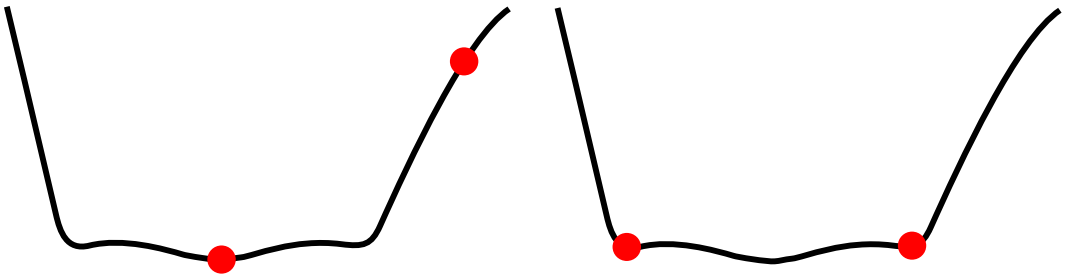
2.3.2 Connection to DivMBest [Bat+12]

DivMBest [Bat+12; PJB14] is a well-known method of obtaining M diverse solutions $\boldsymbol{y}^1, \dots, \boldsymbol{y}^M$ from a single model $E(\boldsymbol{y})$. The approach is very intuitive: the solutions are obtained sequentially; each solution should have good energy and at the same time should be far away from previously obtained solutions. More formally, to get M solutions DivMBest sequentially solves the following optimization problems:

$$\boldsymbol{y}^m = \underset{\boldsymbol{y}\in\mathcal{Y}}{\arg\min} \left[E(\boldsymbol{y}) - \lambda \sum_{i=1}^{m-1} \Delta^{m,i}(\boldsymbol{y}, \boldsymbol{y}^i) \right] \tag{2.3}$$

for $m = 1, 2 \dots, M$, where $\lambda > 0$ determines a trade-off between diversity and energy. Here $\boldsymbol{y}^1$ is the MAP-solution and the function $\Delta^{m,i}\colon L_{\mathcal{V}} \times L_{\mathcal{V}} \to \mathbb{R}$ defines the *diversity* of two labelings. In [Bat+12; PJB14] efficient solvers for (2.3) are proposed for certain diversity measures.

Next, we show that (2.3) is a greedy optimization technique for global multiple diverse solution problem (2.2). The greedy optimization sequentially finds each solution taking into account fixed previously obtained solutions and ignoring yet unknown



(a) Sequentially inferred (b) Jointly inferred

Figure 2.1: Energy landscape with two different couples of solutions depicted by red points. (a) Corresponds to the DivMBest algorithm (2.3), which finds solutions sequentially. (b) Joint inference of diverse solutions (2.2) may lead to lower total energy.

solutions. Let us consider a diversity measure $\Delta^M(\boldsymbol{y}^1, \dots, \boldsymbol{y}^M)$ that can be represented as a sum of diversity functions between all pairs of solutions $\Delta^{i,j}(\boldsymbol{y}^i, \boldsymbol{j}), i > j$:

$$\Delta^M(\boldsymbol{y}^1, \dots, \boldsymbol{y}^M) = \sum_{m=2}^{M} \sum_{i=1}^{m-1} \Delta^{m,i}(\boldsymbol{y}^m, \boldsymbol{y}^i) \tag{2.4}$$

For such diversity measure (2.2) can be rewritten as

$$\underset{(\boldsymbol{y}^1,\dots,\boldsymbol{y}^M)\in\mathcal{Y}^M}{\arg\min} \sum_{m=1}^{M} E(\boldsymbol{y}^m) - \lambda \sum_{m=2}^{M} \sum_{i=1}^{m-1} \Delta^{m,i}(\boldsymbol{y}^m, \boldsymbol{y}^i)\,. \tag{2.5}$$

At step m greedy optimization technique optimizes over terms with $\boldsymbol{y}^m$ only, i.e. $E(\boldsymbol{y}^m)$, $\Delta^{m,i}(\boldsymbol{y}^m, \boldsymbol{y}^i), i < m$, and $\Delta^{k,m}(\boldsymbol{y}^k, \boldsymbol{y}^m), k > m$. The latter terms $\Delta^{k,m}(\boldsymbol{y}^k, \boldsymbol{y}^m), k > m$ are ignored on this step since they contain yet unknown variables $\boldsymbol{y}^k, k > m$. Remaining terms form optimization problem (2.3). Hence, DivMBest is a greedy optimization technique for global diversity optimization problem in the from of (2.5).

Although the DivMBest method (2.3) shows impressive results in a number of computer vision applications [Bat+12; PJB14], we argue that it suffers from its greedy nature. Each new solution is obtained taking into account previously found solutions only, and is not influenced by upcoming solutions. As we show in this work, optimization for all M solutions *jointly* (2.2) allows to improve the resulting solutions. A toy example illustrating our claim is presented in Fig. 2.1. Note that with global diversity optimization problem we do not enforce that the MAP solution is part of the set of solutions. This is in contrast to the DivMBest [Bat+12] method. If this is a requirement then we can run a MAP solver and add its solution to our set.

2.3.3 Connection to DPP [KT10]

Determinantal Point Processes (DPP) [KT10] is another well-known framework to model diversity. It defines a distribution over sets of solutions (objects in DPP's original terminology) so that sets with high quality solutions that are diverse will have high probability. Standard DPP model is defined over sets of all possible sizes. K-DPP

restricts possible set to one specific set size K. More formally, K-DPP distribution is

$$P(\boldsymbol{y}^1, \dots, \boldsymbol{y}^K) = \prod_{k=1}^{K} q(\boldsymbol{y}^k) \times \det S_{\boldsymbol{y}^1,\dots,\boldsymbol{y}^K}\,, \tag{2.6}$$

where $q(\boldsymbol{y}^k)$ determines quality of solution $\boldsymbol{y}^k$ for $k = 1, \dots, K$ and the determinant of specially constructed matrix $S_{\boldsymbol{y}^1,\dots,\boldsymbol{y}^K}$ defines how diverse the set of solutions $\boldsymbol{y}^1, \dots, \boldsymbol{y}^K$ is. Instead of maximizing (2.6), we write down minimization of negative logarithm of (2.6):

$$\operatorname*{arg\,min}_{(\boldsymbol{y}^1,\dots,\boldsymbol{y}^K)\in\mathcal{Y}^K} \sum_{k=1}^{K} -\log q(\boldsymbol{y}^k) - \log\det S_{\boldsymbol{y}^1,\dots,\boldsymbol{y}^K}\,. \tag{2.7}$$

Note, that argmin of (2.7) is equivalent to argmax of (2.6). Defining energy function $E(\boldsymbol{y}^k)$ as negative logarithm of quality function $q(\boldsymbol{y}^k)$, (2.7) has exactly the same form and intuition as general multiple diverse solutions problem (2.2) with the special family of diversity measures defined via determinant. Efficient inference for DPP is possible only for tree-like graphical models. In our work we consider broader family of energy functions. While DPP considers only determinental-based diversity measures, general multiple diverse solutions optimization problem doesn't assume specific form of the diversity measure.

2.4 Formal Problem Definition

Output space for image segmentation tasks has exponential size. There are $L^{H\cdot M}$ possible segmentations in L classes semantic segmentation task for an image with sides of H and W pixels. The general multiple diverse solutions optimization problem (2.2) is NP-hard in the most general case since energy function $E(\boldsymbol{y})$ and diversity measure $\Delta^M(\boldsymbol{y}^1, \dots, \boldsymbol{y}^M)$ can be table functions. Thus, in this section we formally define families of energies and diversity measures that allow efficient optimization. We start from general potential-based energy function definition and then define several useful families of diversity measures.

2.4.1 Energy minimization

In this subsection we formally define energy minimization problem (2.1) for exponential sets of possible solutions. We assume that the energy function is built taking the input into account and consider only output variables $\boldsymbol{y}$ from now on. Let $2^{\mathcal{A}}$ denote the powerset of a set $\mathcal{A}$. The pair $\mathcal{G} = (\mathcal{V}, \mathcal{F})$ is called a *factor graph* and has $\mathcal{V}$ as a finite *set of variable nodes* and $\mathcal{F} \subseteq 2^{\mathcal{V}}$ as *a set of factors*. Each variable node $v \in \mathcal{V}$ is associated with a *variable* y_v taking its values in a finite *set of labels* L_v. The set $L_{\mathcal{A}} = \prod_{v\in\mathcal{A}} L_v$ denotes a Cartesian product of sets of labels corresponding to the subset $\mathcal{A} \subseteq \mathcal{V}$ of variables. Functions $\theta_f \colon L_f \to \mathbb{R}$, associated with factors $f \in \mathcal{F}$, are called *potentials* and define local costs on values of variables and their combinations. The set $\{\theta_f \colon f \in \mathcal{F}\}$ of all potentials is described by $\boldsymbol{\theta}$. For any factor $f \in \mathcal{F}$ the corresponding set of variables $\{y_v \colon v \in f\}$ will be denoted by y_f. *The energy minimization* problem

then consists of finding *a labeling* $\boldsymbol{y}^* = \{y_v \colon v \in \mathcal{V}\} \in L_\mathcal{V}$ which minimizes the total sum of corresponding potentials:

$$\boldsymbol{y}^* = \arg\min_{\boldsymbol{y} \in L_\mathcal{V}} E(\boldsymbol{y}) = \arg\min_{\boldsymbol{y} \in L_\mathcal{V}} \sum_{f \in \mathcal{F}} \theta_f(y_f)\,. \tag{2.8}$$

Problem (2.8) is also known as *MAP-inference*. Labeling $\boldsymbol{y}^*$ satisfying (2.8) will be later called *a solution of the energy-minimization* or *MAP-inference problem*, shortly *MAP-labeling* or *MAP-solution*. Finally, a *model* is defined by the triple $(\mathcal{G}, L_\mathcal{V}, \boldsymbol{\theta})$, i.e. the underlying graph, the sets of labels and the potentials.

2.4.2 Diversity Measure

We formally define families of diversity measures $\Delta^M(\boldsymbol{y}^1, \ldots, \boldsymbol{y}^M)$ we work with. To save space we will further use notation $\{\boldsymbol{y}\}^M$ to define vector of variables $\boldsymbol{y}^1, \ldots, \boldsymbol{y}^M$, *i.e.* $\Delta^M(\{\boldsymbol{y}\}^M) := \Delta^M(\boldsymbol{y}^1, \ldots, \boldsymbol{y}^M)$.

We call diversity measure *node-wise* diversity if it can be represented as

$$\Delta(\{\boldsymbol{y}\}^M) = \sum_{v \in \mathcal{V}} \Delta_v^M(\{y_v\}^M)\,, \tag{2.9}$$

where $\Delta_v^M \colon (L_v)^M \to \mathbb{R}$ is an arbitrary diversity function for node $v \in \mathcal{V}$.

The special case of node-diversity measure is the *node-pair-wise* diversity measure

$$\Delta^M(\{\boldsymbol{y}\}^M) = \sum_{v \in \mathcal{V}} \sum_{i=2}^{M} \sum_{j=1}^{i-1} \Delta_v^{i,j}(y_v^i, y_v^j)\,, \tag{2.10}$$

which, for each node $v \in \mathcal{V}$, is a sum of pairwise factors that connect all pairs of solutions. The special case of this diversity measure is the Hamming distance, i.e.

$$\Delta_v^{i,j}(y, y') = [\![y \neq y']\!]\,, \tag{2.11}$$

where expression $[\![A]\!]$ equals 1 if A is true and 0 otherwise. Note, that Hamming distance is a natural measure of diversity for labeling problems.

An orthogonal property of diversity measures that some optimization techniques require is *permutation-invariance*. We call diversity function permutation-invariant if its value doesn't depend on the order of its operands. Note, that this property is quite natural for function that measure diversity. Order of solutions in a set should not change amount of diversity in the set. We expect most of the reasonable diversity measures to be permutation-invariant. Observe, that Hamming distance is permutation-invariant too.

2.4.3 General Diversity Optimization Problem

We formally define the new general diversity optimization problem (2.2) using factor graph framework as well. We name the new optimization objective as $E^M(\{\boldsymbol{y}\}^M)$:

$$E^M(\{\boldsymbol{y}\}) = \sum_{i=1}^{M} E(\boldsymbol{y}^i) - \lambda \Delta^M(\{\boldsymbol{y}\}^M)\,, \tag{2.12}$$

minimized over $\boldsymbol{y}^1, \dots, \boldsymbol{y}^M \in \mathcal{Y}^M$. The objective (2.12) can be easily represented in the form (2.8) and hence constitutes an energy minimization problem. To achieve this, let us first create M copies $(\mathcal{G}^i, \mathcal{L}^i_{\mathcal{V}}, \boldsymbol{\theta}^i) = (\mathcal{G}, \mathcal{L}_{\mathcal{V}}, \boldsymbol{\theta})$ of the initial model $(\mathcal{G}, \mathcal{L}_{\mathcal{V}}, \boldsymbol{\theta})$. We define the factor-graph $\mathcal{G}^M_1 = (\mathcal{V}^M_1, \mathcal{F}^M_1)$ for the new task as follows. The set of nodes in the new graph is the union of the node sets from the considered copies $\mathcal{V}^M_1 = \bigcup_{i=1}^M \mathcal{V}^i$. Factors are $\mathcal{F}^M_1 = \mathcal{V}^M_1 \cup \bigcup_{i=1}^M \mathcal{F}^i$, i.e. again the union of the initial ones extended by a special factor corresponding to the diversity penalty. Each node $v \in \mathcal{V}^i$ is associated with the label set $L^i_v = L_v$. The corresponding potentials $\boldsymbol{\theta}^M_1$ are defined as $\{-\lambda\Delta^M, \boldsymbol{\theta}^1, \dots, \boldsymbol{\theta}^M\}$. The model $(\mathcal{G}^M_1, \mathcal{L}_{\mathcal{V}^M_1}, \boldsymbol{\theta}^M_1)$ corresponds to the energy (2.12). An optimal M-tuple of these labelings, corresponding to a minimum of (2.12), is a trade-off between low energy of individual labelings $\boldsymbol{y}^i$ and their total diversity.

2.5 Optimization Techniques

In this section we describe previously proposed greedy optimization technique DivMBest [Bat+12] and present several new optimization techniques for the general multiple diverse solution optimization problem (2.12) that impose different constraints on the original energy $E(\boldsymbol{y})$ and diversity measure $\Delta^M(\{\boldsymbol{y}\}^M)$ to be applicable. Fig. 2.2 gives a very general overview of the proposed techniques. We describe each in much more details further in this section. Clique Encoding technique is applicable to the same set of problems as the greedy approach. While it is slower, it outperforms greedy approach in terms of accuracy. Ordering based approach requires diversity measure to be permutation-invariant. This method minimizes (2.12) exactly (if original energy is submodular) and run-time is close to the greedy technique. Parametric-based approach is applicable only to binary submodular energies with additional concavity constraint imposed on the used diversity measure. This technique is an exact minimizer too and it is able to produce solutions faster than the greedy technique.

This overview does not include several high-order diversity measures proposed in [PJB14]. Each of these measure requires a very time-consuming inference technique to use the greedy optimization of (2.12). Moreover, the experimental evaluation in [Kir+15b] suggests that global minimization of (2.12) with node-wise distance diversity measure (2.9) outperforms the greedy optimization with proposed high-order diversity measures.

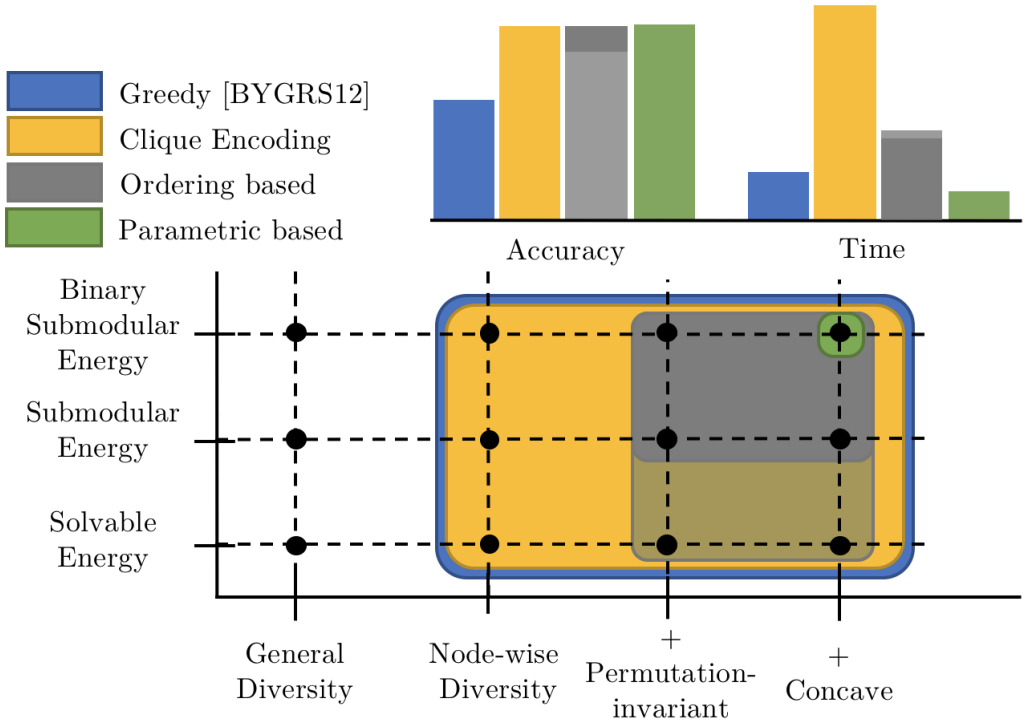


Figure 2.2: Optimization techniques overview for (2.12). Y-axis represents different types of original energy $E(\boldsymbol{y})$ where each next type is a subset of the previous one. Solvable energy is the energy that can be efficiently optimized by an approximate or exact solver. X-axis represents different families of diversity measures. Each next one is a subset of the previous. We will describe the families in more details further in the text.

2.5.1 Greedy Approach: DivMBest [Bat+12]

In what follows we briefly demonstrate how the greedy optimization can be very efficient for (2.12) in case of node-pair-wise diversity measure (2.14). Greedy method subsequently solves for $m = 1, 2 \dots, M$ optimization problems 2.3. We rewrite it here again:

$$\boldsymbol{y}^m = \arg\min_{\boldsymbol{y} \in \mathcal{Y}} \left[E(\boldsymbol{y}) - \lambda \sum_{i=1}^{m-1} \Delta^{m,i}(\boldsymbol{y}, \boldsymbol{y}^i) \right] \tag{2.13}$$

If $\Delta^{m,i}(\boldsymbol{y}, \boldsymbol{y}^i)$ is represented by a sum of node-wise diversity measures $\Delta_v \colon L_v \times L_v \to \mathbb{R}$,

$$\Delta(\boldsymbol{y}, \boldsymbol{y}') = \sum_{v \in \mathcal{V}} \Delta_v(y_v, y'_v) \ , \tag{2.14}$$

then the diversity potentials are split to a sum of *unary* potentials, i.e. those associated with additional factors $\{v\}$, $v \in \mathcal{V}$. This implies that in case efficient graph-cut based inference methods (including α-expansion [BVZ01], α-β-swap [BVZ01] or their generalizations [Aro+15; Fix+11]) are applicable to the initial problem (2.8) then they remain applicable to the augmented problem (2.13), which assures efficiency of the

method.

2.5.2 Clique Encoding

In this section we propose new solver for general multiple diverse solutions optimization (2.12) with a node-wise diversity measure (2.9). We show that if the original optimization problem (2.1) was (approximately) solvable with α-expansion or α-β-swap[BJ01] our model, delivering M best diverse solutions, maintains this property.

Objective (2.12) with a node-wise diversity measure (2.9) reads as follows:

$$E^M(\{\boldsymbol{y}\}) = \sum_{i=1}^{M} E(\boldsymbol{y}^i) - \lambda \sum_{v \in \mathcal{V}} \Delta_v^M(\{y_v\}^M) \tag{2.15}$$

We now present an alternative representation of the model (2.15). This representation has fewer number of nodes but at the same time a larger label space. We will see that this representation is easier to optimize. Expanding energy function $E(\boldsymbol{y})$ as a sum of potentials (2.8), the energy (2.15) can be rewritten as

$$E^M(\{\boldsymbol{y}\}) = \sum_{i=1}^{M} \left[\sum_{\substack{f \in \mathcal{F} \\ |f|=1}} \theta_f(y_f^i) + \sum_{\substack{f \in \mathcal{F} \\ |f|>1}} \theta_f(y_f^i) \right] - \lambda \sum_{v \in \mathcal{V}} \Delta_v^M(\{y_v\}^M)\,. \tag{2.16}$$

Assume w.l.o.g. that $\{v\} \in \mathcal{F}$ for all $v \in \mathcal{V}$. Then we denote *unary* potentials θ_f for $|f| = 1$ as θ_v and regrouping terms, the above equation can be written as

$$\sum_{v \in \mathcal{V}} \left[\sum_{i=1}^{M} \theta_v(y_v^i) - \lambda \Delta_v^M(\{y_v\}^M) \right] + \sum_{\substack{f \in \mathcal{F} \\ |f|>1}} \sum_{i=1}^{M} \theta_f(y_f^i)\,.$$

Let us introduce the new variables $\boldsymbol{z}_v = (y_v^1, \dots, y_v^M)$, $v \in \mathcal{V}$ and the respective label sets $\hat{L}_v = (L_v)^M$. Informally, each label of a new variable $\boldsymbol{z}_v$ in a node v corresponds to an M-tuple of labels from the original task. In other words, we simply enumerate all possible label combinations in each node v, that are possible by M solutions. The new potentials $\hat{\theta}_v \colon \hat{L}_v \to \mathbb{R}$, $v \in \mathcal{V}$ and $\hat{\theta}_f \colon (L_f)^M \to \mathbb{R}$, $f \in \mathcal{F} \colon |f| > 1$ are defined as

$$\hat{\theta}_v(\boldsymbol{z}_v) = \sum_{i=1}^{M} \theta_v(y_v^i) - \lambda \Delta_v^M(\{y_v\}^M)\,, \tag{2.17}$$

$$\hat{\theta}_f(\boldsymbol{z}_f) = \sum_{i=1}^{M} \theta_f(y_f^i)\,. \tag{2.18}$$

In this notation the energy is given as

$$E^M(\{\boldsymbol{y}\}) = \sum_{v \in \mathcal{V}} \hat{\theta}_v(\boldsymbol{z}_v) + \sum_{\substack{f \in \mathcal{F} \\ |f|>1}} \hat{\theta}_f(\boldsymbol{z}_f)\,. \tag{2.19}$$

Pairwise model. For second order models (i.e. the cardinality of factors is two at most) equation (2.19) is written as

$$E^M(\{\boldsymbol{y}\}) = \sum_{v\in\mathcal{V}} \hat{\theta}_v(\boldsymbol{z}_v) + \sum_{uv\in\mathcal{F}} \hat{\theta}_{uv}(\boldsymbol{z}_u, \boldsymbol{z}_v)\,. \tag{2.20}$$

The following Theorem 1 basically states that in case the original MAP-inference problem is (approximately) solvable with α-β-swap [BVZ01] (α-expansion [BVZ01]) then minimization of $E^M(\{\boldsymbol{y}\})$ in (2.20) can be performed with α-β swap (α-expansion) as well.

Definition 1. *For any set L the function $f\colon L\times L\to\mathbb{R}$ is called* a semi-metric *if for all $x, x'\in L$ there holds: (i) $f(x,x')\geq 0$; (ii) $f(x,x')=0$ iff $x=x'$; (iii) $f(x,x')=f(x',x)$.*

Definition 2. *Function $f\colon L\times L\to\mathbb{R}$ is called* a metric *if it is a semi-metric and additionally there holds:*
$f(x,x')+f(x',x'')\geq f(x,x'')$, $\forall x,x',x''\in L$.

Theorem 1. *Let $L_v=L_u$, $uv\in\mathcal{F}$ and functions θ_{uv} be semi-metrics (metrics). Then functions $\hat{\theta}_{uv}(\boldsymbol{z}_u,\boldsymbol{z}_v)$ defined as in* (2.18) *are semi-metrics (metrics) as well.*

Proof. Let $y_v^i\in L$, $v\in\mathcal{V}$ and $i=1,\dots,M$ be arbitrary $|\mathcal{V}||M|$ labels. Let $\boldsymbol{z}_v$ be defined as $\boldsymbol{z}_v=(y_v^1,\dots,y_v^M)$ like in Section 2.6.2. We show that if conditions of Definitions 1 and 2 hold for θ_{uv}, $uv\in\mathcal{E}$, then they hold for $\hat{\theta}_{uv}$ as well: (i) Summing up $\theta_{uv}(y_u^i,y_v^i)\geq 0$ over $i=1,\dots,M$ gives that

$$\hat{\theta}_{uv}(\boldsymbol{z}_u,\boldsymbol{z}_v)=\sum_{i=1}^{M}\theta_{uv}(y_u^i,y_v^i)\geq 0$$

(ii) From $\theta_{uv}(y_u^i,y_v^i)=0$ iff $y_u^i=y_v^i$ and $\theta_{uv}(y_u^i,y_v^i)\geq 0$ otherwise, follows that

$$\hat{\theta}_{uv}(\boldsymbol{z}_u,\boldsymbol{z}_v)=\sum_{i=1}^{M}\theta_{uv}(y_u^i,y_v^i)=0$$

iff $\boldsymbol{z}_u=\boldsymbol{z}_v$. (iii) Summing up $\theta_{uv}(y_u^i,y_v^i)=\theta_{uv}(y_v^i,y_u^i)$ over $i=1,\dots,M$ gives that

$$\hat{\theta}_{uv}(\boldsymbol{z}_u,\boldsymbol{z}_v)=\sum_{i=1}^{M}\theta_{uv}(y_u^i,y_v^i)=\sum_{i=1}^{M}\theta_{uv}(y_v^i,y_u^i)=\hat{\theta}_{uv}(\boldsymbol{z}_v,\boldsymbol{z}_u)\,.$$

(iv) Inequality $\theta_{uv}(y_u^i,s^i)+\theta_{uv}(s^i,y_v^i)\geq\theta_{uv}(y_u^i,y_v^i)$ holds for any $s^i\in L$ and $i=1,\dots,M$ according to Definition 2. Summing it up over i gives that

$$\sum_{i=1}^{M}\big(\theta_{uv}(y_u^i,s^i)+\theta_{uv}(s^i,y_v^i)\big)\geq\underbrace{\sum_{i=1}^{M}\theta_{uv}(y_u^i,y_v^i)}_{\hat{\theta}_{uv}(\boldsymbol{z}_u,\boldsymbol{z}_v)} \tag{2.21}$$

The left-hand side of (2.21) can be rewritten as

$$\sum_{i=1}^{M} \theta_{uv}(y_u^i, s^i) + \sum_{i=1}^{M} \theta_{uv}(s^i, y_v^i) = \hat{\theta}_{uv}(\boldsymbol{z}_u, \boldsymbol{s}) + \hat{\theta}_{uv}(\boldsymbol{s}, \boldsymbol{z}_v)\,, \tag{2.22}$$

where $\boldsymbol{s}$ denotes $(s^1, \dots, s^M)$.

Plugging (2.22) back to (2.21) finalizes the proof. □

For instance, in the special case of *Potts model* $\theta_{uv}(y, y') = [\![y \neq y']\!]$ the pairwise factors defined by (2.18) constitute the Hamming distance between vectors $\boldsymbol{z}_v$ representing the new labels:

$$\hat{\theta}_{uv}(\boldsymbol{z}_u, \boldsymbol{z}_v) := \sum_{i=1}^{M} \theta_{uv}(y_u^i, y_v^i) = \sum_{i=1}^{M} [\![y_u^i \neq y_v^i]\!]\,. \tag{2.23}$$

Both Potts potentials and Hamming distance are metrics, which defines a special case of Theorem 1.

K-truncated Clique Encoding. The disadvantage of the clique encoding representation (2.19) is an exponential growth of cardinality of the label set $\hat{L}_v = (L_v)^M$, which implies inefficiency for inference with large L_v and especially a large M. For these cases we propose an efficient approximative algorithm combining clique encoding (2.19) and greedy minimization for the energy (2.12). Though it can be used with the node-diversity measures (2.9) we describe it for the special case of the node-par-wise diversities (2.14), as it is used in our experiments. The pseudo-code for the K-Truncated Clique Encoding algorithm can be written as follows

Algorithm 1 K-truncated Clique Encoding

Require: $(\mathcal{G},\ L_{\mathcal{V}}, \boldsymbol{\theta})$ – original model,
$\lambda \in \mathbb{R}$ – diversity parameter,
$M \in \mathbb{N}$ – total number of diverse labelings,
$K < M$ – num. of processed labelings in each step.

1: **for** $i = 0, \dots, \lfloor \frac{M}{K} \rfloor$ **do**
2: $\quad s = iK + 1; \quad t = \min\{M, (i+1)K\}$
3: $\quad \{\boldsymbol{y}^s, \dots, \boldsymbol{y}^t\} = \arg \min_{\{\boldsymbol{x}^s, \dots, \boldsymbol{x}^t\}} \Big[E^K(\boldsymbol{x}^s, \dots, \boldsymbol{x}^t) - \lambda \sum_{v \in \mathcal{V}} \sum_{l=s}^{t} \sum_{m=1}^{s-1} \Delta_v(x_v^l, y_v^m) \Big]$
4: **end for**
5: **return** $\{\boldsymbol{y}^1, \dots, \boldsymbol{y}^M\}$

In each iteration the algorithm performs optimization with respect to at most K labelings $\{\boldsymbol{y}^s, \dots, \boldsymbol{y}^t\}$, $t - s + 1 = K$, (less than K in the last iteration, if M is not dividable by K) given already computed labelings $\{\boldsymbol{y}^1, \dots, \boldsymbol{y}^{s-1}\}$. Diversity of $\{\boldsymbol{y}^s, \dots, \boldsymbol{y}^t\}$ with respect to $\{\boldsymbol{y}^1, \dots, \boldsymbol{y}^{s-1}\}$ is provided by taking into account the sum of corresponding diversity terms $\lambda \sum_{v \in \mathcal{V}} \sum_{l=s}^{t} \sum_{m=1}^{s-1} \Delta_v(x_v^l, y_v^m)$ playing the role of addition

to unary potentials. Minimization (possibly approximate) in the algorithm is done with the clique encoding approach (2.19).

Overall, algorithm performs a greedy optimization similar to DivMBest (2.3) with the difference that in each iteration K labelings are inferred *jointly* instead of a single one. The method coincides with DivMBest (2.3) for $K = 1$ and with clique encoding for $K = M$. As it is shown in [Kir+15a], the K-Trunctaed Clique Encoding algorithm significantly outperforms DivMBest (2.3) already for $K = 2$. Larger values of K lead to further improvements.

2.5.3 Ordering Based Approach

In this section we present ordering based approach:

- We show that exact solution for minimization of objective (2.15) with a binary submodular original energy $E(\boldsymbol{y})$ can be found by solving a submodular optimization, and hence can be very efficient for *any* node-wise diversity measure.
- We demonstrate that for certain diversity measures, such as *e.g*. Hamming distance, exact minimizer of $E^M(\{\boldsymbol{y}\}^M)$ (2.15) with a multilabel submodular energy $E(\boldsymbol{y})$ can be found by solving a submodular MAP-inference problem, which also implies applicability of efficient graph cut-based solvers.
- We give the insight that if the $E(\boldsymbol{y})$ is submodular then the exact solution of $E^M(\{\boldsymbol{y}\}^M)$ (2.15) minimization can be always fully ordered with respect to the natural partial order, induced in the space of all solutions.
- We show experimentally that if $E(\boldsymbol{y})$ is submodular, the new method is quantitatively at least as good as clique encoding approach proposed in the previous section and is considerably better than DivMBest [Bat+12]. The main advantage is a major speed up over clique encoding, up to the order of two magnitudes. New method has the same order of magnitude run-time as [Bat+12].
- Ordering based approach can be applied to a non-submodular energy $E(\boldsymbol{y})$ too. Its results are slightly inferior to clique encoding, but the advantage with respect to gain in speed up still remains.

Submodularity. We start from formally defining submodular energies. In what follows we will assume that the sets L_v, $v \in \mathcal{V}$, of labels are completely ordered. This implies that for any $s, t \in \mathcal{L}_v$ their maximum and minimum, denoted as $s \vee t$ and $s \wedge t$ respectively, are well-defined. Similarly let $\boldsymbol{y}_1 \vee \boldsymbol{y}_2$ and $\boldsymbol{y}_1 \wedge \boldsymbol{y}_2$ denote the node-wise maximum and minimum of any two labelings $\boldsymbol{y}_1, \boldsymbol{y}_2 \in L_{\mathcal{A}}$, $\mathcal{A} \subseteq \mathcal{V}$. Potential θ_f is called *submodular*, if for any two labelings $\boldsymbol{y}_1, \boldsymbol{y}_2 \in L_f$ it holds[1]:

$$\theta_f(\boldsymbol{y}_1) + \theta_f(\boldsymbol{y}_2) \geq \theta_f(\boldsymbol{y}_1 \vee \boldsymbol{y}_2) + \theta_f(\boldsymbol{y}_1 \wedge \boldsymbol{y}_2)\,. \tag{2.24}$$

Potential θ will be called *supermodular*, if $(-\theta)$ is submodular.

[1]Pairwise binary potentials satisfying $\theta_f(0,1) + \theta_f(1,0) \geq \theta_f(0,0) + \theta_f(1,1)$ build an important special case of this definition.

Energy E is called submodular if for any two labelings $\boldsymbol{y}_1, \boldsymbol{y}_2 \in L_{\mathcal{V}}$ it holds:

$$E(\boldsymbol{y}_1) + E(\boldsymbol{y}_2) \geq E(\boldsymbol{y}_1 \vee \boldsymbol{y}_2) + E(\boldsymbol{y}_1 \wedge \boldsymbol{y}_2)\,. \tag{2.25}$$

Submodularity of energy trivially follows from the submodularity of all its non-unary potentials θ_f, $f \in \mathcal{F}$, $|f| > 1$. In the pairwise case the inverse also holds: submodularity of energy implies also submodularity of all its (pairwise) potentials (e.g. [Wer07, Thm. 12]). There are efficient methods for solving energy minimization problems with submodular potentials, based on its transformation into min-cut/max-flow problem [KZ04; SF06; Ish03] in case all potentials are either unary or pairwise or to a submodular max-flow problem in the higher-order case [Kol12; Fix+11; Aro+15].

Ordered M solutions. In what follows we will write $\boldsymbol{z^1} \leq \boldsymbol{z^2}$ for any two vectors $\boldsymbol{z}^1$ and $\boldsymbol{z^2}$ meaning that the inequality holds coordinate-wise.

For an arbitrary set $\mathcal{A}$ we will call a function $f\colon (\mathcal{A})^n \to \mathbb{R}$ of n variables *permutation invariant* if for any $(x^1, x^2, \ldots, x^n) \in (\mathcal{A})^n$ and any permutation π it holds $f(x^1, x^2, \ldots, x^n) = f(x^{\pi(1)}, x^{\pi(2)}, \ldots, x^{\pi(n)})$. In what follows we will consider mainly permutation invariant diversity measures.

Let us consider two arbitrary labelings $\boldsymbol{y}^1, \boldsymbol{y}^2 \in L_{\mathcal{V}}$ and their node-wise minimum $\boldsymbol{y}^1 \wedge \boldsymbol{y}^2$ and maximum $\boldsymbol{y}^1 \vee \boldsymbol{y}^2$. Since $(y^1_v \wedge y^2_v, y^1_v \vee y^2_v)$ is either equal to (y^1_v, y^2_v) or to (y^2_v, y^1_v), for any permutation invariant node diversity measure it holds $\Delta^2_v(y^1_v, y^2_v) = \Delta^2_v(y^1_v \wedge y^2_v, y^1_v \vee y^2_v)$. This in its turn implies $\Delta^2(\boldsymbol{y}^1 \wedge \boldsymbol{y}^2, \boldsymbol{y}^1 \vee \boldsymbol{y}^2) = \Delta^2(\boldsymbol{y}^1, \boldsymbol{y}^2)$ for any node-wise diversity measure of the form (2.9). If E is submodular, then from (2.25) it additionally follows that

$$E^2(\boldsymbol{y}^1 \wedge \boldsymbol{y}^2, \boldsymbol{y}^1 \vee \boldsymbol{y}^2) \leq E^2(\boldsymbol{y}^1, \boldsymbol{y}^2)\,, \tag{2.26}$$

where E^2 is defined as in (2.12). Note, that $(\boldsymbol{y}^1 \wedge \boldsymbol{y}^2) \leq (\boldsymbol{y}^1 \vee \boldsymbol{y}^2)$. Generalizing these considerations to M labelings one obtains

Theorem 2. *Let E be submodular and Δ^M be a node-wise diversity measure with each component Δ^M_v being permutation invariant. Then there exists an ordered M-tuple $(\boldsymbol{y}^1, \ldots, \boldsymbol{y}^M)$, $\boldsymbol{y}^i \leq \boldsymbol{y}^j$ for $1 \leq i < j \leq M$, such that for any $(\boldsymbol{z}^1, \ldots, \boldsymbol{z}^M) \in (L_{\mathcal{V}})^M$ it holds*

$$E^M(\{\boldsymbol{y}\}) \leq E^M(\{\boldsymbol{z}\})\,, \tag{2.27}$$

where E^M is defined as in (2.12)*.*

Proof. Let us consider the operation $\texttt{order}(\{\boldsymbol{y}\}, i, j)$, which takes a set of labelings $\{\boldsymbol{y}\} \in (L_{\mathcal{V}})^M$, two indices $i < j \in 1, \ldots, M$ and replaces labelings $\boldsymbol{y}^i$ and $\boldsymbol{y}^j$ by their node-wise minimum $\boldsymbol{y}^i \wedge \boldsymbol{y}^j$ and maximum $\boldsymbol{y}^i \vee \boldsymbol{y}^j$ respectively. As a result, this operation returns the new set of labelings:

$$(\boldsymbol{y}^1, \ldots, \boldsymbol{y}^{i-1}, \boldsymbol{y}^i \wedge \boldsymbol{y}^j, \boldsymbol{y}^{i+1}, \ldots, \boldsymbol{y}^{j-1}, \boldsymbol{y}^i \vee \boldsymbol{y}^j, \boldsymbol{y}^{j+1}, \ldots, \boldsymbol{y}^M). \tag{2.28}$$

In what follows we will show that

$$E^M\left(\texttt{order}(\{\boldsymbol{y}\}, i, j)\right) \leq E^M(\{\boldsymbol{y}\})\,. \tag{2.29}$$

Let $\{\boldsymbol{y}'\} = \texttt{order}(\{\boldsymbol{y}\}, i, j)$. Then $\{\boldsymbol{y}'\}_v$ is equal either to $(y_v^1, \dots, y_v^i, \dots, y_v^j, \dots, y_v^M)$ or to $(y_v^1, \dots, y_v^j, \dots, y_v^i, \dots, y_v^M)$. Since each Δ_v^M is permutation invariant, $\Delta^M(\{\hat{\boldsymbol{y}}'\}) = \Delta^M(\{\hat{\boldsymbol{y}}\})$. Summing it up with the following inequality, which follows from the submodularity of E,

$$\sum_{k=1}^{M} E(\boldsymbol{y}'^k) = \sum_{\substack{k=1 \\ k\neq i, k\neq j}}^{M} E(\boldsymbol{y}^k) + E(\boldsymbol{y}^i \wedge \boldsymbol{y}^j) + E(\boldsymbol{y}^i \vee \boldsymbol{y}^j) \leq \sum_{k=1}^{M} E(\boldsymbol{y}^k). \tag{2.30}$$

one obtains (2.29).

Assume the set of labelings $\{\hat{\boldsymbol{y}}\} = (\hat{\boldsymbol{y}}^1, \dots, \hat{\boldsymbol{y}}^M)$ is a solution to (2.12):

$$\{\hat{\boldsymbol{y}}\} = \arg\min_{\{\boldsymbol{y}\}} E^M(\{\boldsymbol{y}\}). \tag{2.31}$$

Let us iteratively apply the operation $\{\hat{\boldsymbol{y}}\} := \texttt{order}(\{\hat{\boldsymbol{y}}\}, i, j)$ such, that indexes i and j follow the bubble-sort algorithm [Cor09]. Each operation performs sorting for a single pair $i < j$ of indexes and due to (2.29) the energy $E^M\{\hat{\boldsymbol{y}}\}$ does not increase after the operation. As a result of the algorithm we obtain the ordered labeling set $\{\hat{\boldsymbol{y}}\}$ satisfying

$$E^M(\{\hat{\boldsymbol{y}}\}) \leq \min_{\{\boldsymbol{y}\}} E^M(\{\boldsymbol{y}\}), \tag{2.32}$$

which finalizes our proof. □

Theorem 2 in particular claims that in the binary case $L_v = \{0, 1\}$, $v \in \mathcal{V}$, the optimal M labelings define nested subsets of nodes, corresponding to the label 1.

Submodular formulation of general multiple diverse solutions problem. Due to Theorem 2, for submodular energies and node-wise diversity measures it is sufficient to consider only ordered M-tuples of labelings.

This order can be enforced by modifying the diversity measure accordingly:

$$\hat{\Delta}_v^M(\{y_v\}^M) := \begin{cases} \Delta_v^M(\{y_v\}^M), & y^1 \leq y^2 \leq \dots \leq y^M \\ -\infty, & \text{otherwise} \end{cases}, \tag{2.33}$$

and using it instead of the initial measure Δ_v^M. Note that $\hat{\Delta}_v^M$ is *not* permutation invariant. In practice one can use sufficiently big numbers in place of ∞ in (2.33). This implies

Lemma 1. *Let E be submodular and Δ^M be a node-wise diversity measure with each component Δ_v^M being permutation invariant. Then any solution of* the ordering enforcing *M-best-diverse problem*

$$\hat{E}^M(\{\boldsymbol{y}\}) = \sum_{i=1}^{M} E(\boldsymbol{y}^i) - \lambda \sum_{v\in\mathcal{V}} \hat{\Delta}_v^M(\{y_v\}^M) \tag{2.34}$$

is a solution of the corresponding M-best-diverse problem (2.12)

$$E^M(\{\boldsymbol{y}\}) = \sum_{i=1}^{M} E(\boldsymbol{y}^i) - \lambda \sum_{v \in \mathcal{V}} \Delta_v^M(\{y_v\}^M)\,, \tag{2.35}$$

where $\hat{\Delta}_v^M$ and Δ_v^M are related by (2.33)*.*

Proof. Since E is submodular and each Δ_v^M is permutation invariant we can apply Theorem 2 for E^M. This implies that E^M has an ordered minimizer $\{\boldsymbol{y}^*\}$ and $\hat{E}^M(\{\boldsymbol{y}^*\}) = E^M(\{\boldsymbol{y}^*\})$.

Since the diversity controlling parameter $\lambda > 0$, the value of $-\lambda\hat{\Delta}_v^M(y^1, \dots, y^M)$ is equal to $+\infty$ for an unordered set $(\boldsymbol{y}^1, \dots, \boldsymbol{y}^M)$. Therefore, $\hat{E}^M(\{\boldsymbol{y}\})$ can be represented as follows:

$$\hat{E}^M(\{\boldsymbol{y}\}) = \begin{cases} E^M(\{\boldsymbol{y}\}), & \boldsymbol{y}^1 \leq \boldsymbol{y}^2 \leq \dots \leq \boldsymbol{y}^M \\ \infty, & \text{otherwise} \end{cases}. \tag{2.36}$$

This implies $\arg\min_{\{y\}} \hat{E}^M(\{\boldsymbol{y}\}) \subseteq \arg\min_{\{y\}} E^M(\{\boldsymbol{y}\})$, which finalizes the proof. □

We will say that a vector $(y^1, \dots, y^M) \in (L_v)^M$ is *ordered*, if it holds $y^1 \leq y^2 \leq \dots \leq y^M$.

Given submodularity of E the submodularity (an hence – solvability) of E^M in (2.35) would trivially follow from the supermodularity of Δ^M. However there hardly exist supermodular diversity measures. The ordering provided by Theorem 2 and the corresponding form of the ordering-enforcing diversity measure $\hat{\Delta}^M$ significantly weaken this condition, which is precisely stated by the following lemma. In the lemma we substitute ∞ of (2.33) with a sufficiently big values such as $C_\infty \geq \max_{\{\boldsymbol{y}\}} E^M(\{y\})$ for the sake of numerical implementation. Moreover, this values will differ from each other to keep $\hat{\Delta}_v^M$ supermodular.

Lemma 2. *Let for any two* **ordered** *vectors $\boldsymbol{y} = (y^1, \dots, y^M) \in (L_v)^M$ and $\boldsymbol{z} = (z^1, \dots, z^M) \in (L_v)^M$ it holds*

$$\Delta_v(\boldsymbol{y} \vee \boldsymbol{z}) + \Delta_v^M(\boldsymbol{y} \wedge \boldsymbol{z}) \geq \Delta_v^M(\boldsymbol{y}) + \Delta_v(\boldsymbol{z}), \tag{2.37}$$

where $\boldsymbol{y} \vee \boldsymbol{z}$ and $\boldsymbol{y} \wedge \boldsymbol{z}$ are element-wise maximum and minimum respectively. Then $\hat{\Delta}_v^M$, defined as

$$\Delta_v^M(\{y_v\}^M) - C_\infty \cdot \left[\sum_{i=1}^{M-1} \sum_{j=i+1}^{M} 3^{\max(0, y^i - y^j)} - 1 \right] \tag{2.38}$$

is supermodular.

Proof. Let us consider $f(\boldsymbol{y}) = -\sum_{i=1}^{M} \sum_{j=i+1}^{M} \left(3^{\max(0, y^i - y^j)} - 1\right)$. This potential is a sum of pairwise potentials $f_{ij}(y^i, y^j) = -\left(3^{\max(0, y^i - y^j)} - 1\right)$. They are supermodular,

which can be checked directly by definition. Moreover, by construction

$$f(\boldsymbol{y} \vee \boldsymbol{z}) + f(\boldsymbol{y} \wedge \boldsymbol{z}) = f(\boldsymbol{y}) + f(\boldsymbol{z}) \tag{2.39}$$

if either (i) both $\boldsymbol{y}$ and $\boldsymbol{z}$ are ordered vectors or (ii) $\boldsymbol{y}$ and $\boldsymbol{z}$ are comparable, i.e. $(\boldsymbol{y} \vee \boldsymbol{z}, \boldsymbol{y} \wedge \boldsymbol{z})$ is either equal to $(\boldsymbol{y}, \boldsymbol{z})$ or to $(\boldsymbol{y}, \boldsymbol{z})$. Let us verify supermodularity of (2.38) by definition, i.e. for any $\boldsymbol{y} \in (L_v)^M$ and $\boldsymbol{z} \in (L_v)^M$, the following inequality has to be satisfied:

$$\hat{\Delta}_v^M(\boldsymbol{y} \vee \boldsymbol{z}) + \hat{\Delta}_v^M(\boldsymbol{y} \wedge \boldsymbol{z}) \geq \hat{\Delta}_v^M(\boldsymbol{y}) + \hat{\Delta}_v^M(\boldsymbol{z}). \tag{2.40}$$

For any ordered $\boldsymbol{y} \in (L_v)^M$ it holds $f(\boldsymbol{y}) = 0$. Therefore, taking into account (2.37), the inequality (2.40) holds for any ordered $\boldsymbol{y}$ and $\boldsymbol{z}$. For any comparable $\boldsymbol{y}$ and $\boldsymbol{z}$ the inequality (2.40) is trivial. For any other $\boldsymbol{y}$ and $\boldsymbol{z}$ the following strict inequality holds $f(\boldsymbol{y} \vee \boldsymbol{z}) + f(\boldsymbol{y} \wedge \boldsymbol{z}) > f(\boldsymbol{y}) + f(\boldsymbol{z})$. This implies that for a sufficiently big C_∞, the inequality (2.40) holds for arbitrary $\Delta_v(y^1, \dots, y^M)$. □

Note, eq. (2.33) and (2.38) are the same up to the infinity values in (2.33). Though condition (2.37) resembles the supermodularity condition, it has to be fulfilled for *ordered* vectors only. The following corollaries of Lemma 2 give two most important examples of the diversity measures fulfilling (2.37).

Corollary 1. *Let $|L_v| = 2$ for all $v \in \mathcal{V}$. Then the statement of Lemma 2 holds for* arbitrary $\Delta_v \colon (L_v)^M \to \mathbb{R}$.

Corollary 2. *Let $\Delta_v^M(\{y_v\}^M) = \sum_{i=1}^{M-1} \sum_{j=i+1}^{M} \Delta^{i,j}(y^i, y^j)$. Then the condition of Lemma 2 is equivalent to*

$$\Delta^{i,j}(y^i, y^j) + \Delta^{i,j}(y^i{+}1, y^j{+}1) \geq \Delta^{i,j}(y^i{+}1, y^j) + \Delta^{i,j}(y^i, y^j{+}1) \text{ for } y^i < y^j \tag{2.41}$$

and $1 \leq i < j \leq M$.

In particular, condition (2.41) *is satisfied for the Hamming distance $\Delta^{i,j}(y, y') = [\![y \neq y']\!]$.*

The following theorem trivially summarizes Lemmas 1 and 2:

Theorem 3. *Let energy E and diversity measure Δ^M satisfy conditions of Lemmas 1 and 2. Then the ordering enforcing problem* (2.34) *delivers solution to the M-best-diverse problem* (2.35) *and is submodular. Moreover, submodularity of all non-unary potentials of the energy E implies submodularity of all non-unary potentials of the ordering enforcing energy $\hat{E}^M$.*

Proof. Since energy E and diversity measure Δ^M satisfy conditions of Lemma 1, the ordering enforcing problem (2.34) delivers solution to the M-best-diverse problem (2.35). Moreover, since each component Δ_v^M of Δ^M satisfies conditions of Lemma 2, the function $\hat{\Delta}^M$ is supermodular and $-\hat{\Delta}^M$ is submodular. Since energy E is submodular either, the ordering enforcing energy $\hat{E}^M$ is submodular as sum of submodular functions. □

The theorem shows that under conditions of Lemmas 1 and 2 an exact solution of (2.15) can be found by solving a submodular problem (2.34). Hence, exact solution can be found in polynomial time.

2.5.4 Parametric based Approach

Submodularity of original energy $E(\boldsymbol{y})$ allows us to find exact solutions of (2.15) by solving submodular minimization (2.34). While delivering exact solution, the optimization technique can be still slower than DivMBest [Bat+12]. In this section, we show that it is possible to find exact solution faster than DivMBest [Bat+12] finds an approximate solution if original energy $E(\boldsymbol{y})$ is binary and submodular.

As we show in the previous section, for binary submodular energies $E(\boldsymbol{y})$ exact solution of general multiple diverse minimization problem forms nested set $\boldsymbol{y}^1, \dots, \boldsymbol{y}^M$; the same property holds for solutions of well-known *Parametric Submodular Minimization* [GGT89; Hoc08; FI03]. Exploring this similarity, we present *a closed-form formula* for the parameters values, which corresponds to the exact solution. The values can be computed in advance, prior to any optimization, which allows to obtain each solution independently.

Our theoretical results suggest a number of efficient algorithms for the problem. We describe two simplest of them, sequential and parallel. Both are considerably faster than the popular technique [Bat+12] and are as easy to implement.

Permutation-invariant node-wise diversity measure. In this section we will use only node-wise diversity measures (2.9). Moreover, we will stick to *permutation-invariant* diversity measures. In other words, such measures that $\Delta_v^M(\{y_v\}^M) = \Delta_v^M(\pi(\{y_v\}))$ for any permutation π of variables $\{y_v\}$.

Let the expression $[\![A]\!]$ be equal to 1 if A is true and 0 otherwise. Let also $m_v^0 = \sum_{m=1}^M [\![y_v^m = 0]\!]$ count the number of 0's in $\{\boldsymbol{y}_v\}$. In the binary case $L_v = \{0, 1\}$, any permutation invariant measure can be represented as

$$\Delta_v^M(\{y_v\}) = \bar{\Delta}_v^M(m_v^0)\,. \tag{2.42}$$

To keep notation simple, we will use Δ_v^M for both representations: $\Delta_v^M(\{\boldsymbol{y}\}_v)$ and $\bar{\Delta}_v^M(m_v^0)$.

Example 1 (Hamming distance diversity)**.** *Consider the common node diversity measure, the sum of Hamming distances between each pair of labels:*

$$\Delta_v^M(\{y_v\}^M) = \sum_{i=1}^{M} \sum_{j=i+1}^{M} [\![y_v^i \neq y_v^j]\!]. \tag{2.43}$$

This measure is permutation invariant. Therefore, it can be written as a function of the number m_v^0:

$$\Delta_v^M(m_v^0) = m_v^0 \cdot (M - m_v^0). \tag{2.44}$$

Parametric submodular minimization. Let $\boldsymbol{\gamma} \in \mathbb{R}^{|\mathcal{V}|}$, $i = \{1, \dots, k\}$ be a *vector of parameters* with the coordinates indexed by the node index $v \in \mathcal{V}$. We define the *parametric energy minimization* as the problem of evaluating the function

$$\min_{\boldsymbol{y} \in L_{\mathcal{V}}} E^{\boldsymbol{\gamma}}(\boldsymbol{y}) := \min_{\boldsymbol{y} \in L} \left[E(\boldsymbol{y}) + \sum_{v \in \mathcal{V}} \gamma_v y_v \right] \tag{2.45}$$

for all values of the parameter $\boldsymbol{\gamma} \in \mathit{\Gamma} \subseteq \mathbb{R}^{|\mathcal{V}|}$. The most important cases of the parametric energy minimization are

- the monotonic parametric max-flow problem [GGT89; Hoc08], which corresponds to the case when E is a binary submodular pairwise energy and $\mathit{\Gamma} = \{\nu \in \mathbb{R}^{|\mathcal{V}|} \colon \nu_v = \gamma_v(\lambda)\}$ and functions $\gamma_v \colon \mathit{\Lambda} \to \mathbb{R}$ are non-increasing for $\mathit{\Lambda} \subseteq \mathbb{R}$.
- a subclass of the parametric submodular minimization [FI03; Bac13], where E is submodular and $\mathit{\Gamma} = \{\boldsymbol{\gamma}^1, \boldsymbol{\gamma}^2, \dots, \boldsymbol{\gamma}^k \in \mathbb{R}^{|\mathcal{V}|} \colon \boldsymbol{\gamma}^1 \geq \boldsymbol{\gamma}^2 \geq \dots \geq \boldsymbol{\gamma}^k\}$, where operation $\geq$ is applied coordinate-wise.

It is known [Top78] that in these two cases, (i) the highest minimizers $\boldsymbol{y}^1, \dots, \boldsymbol{y}^k \in L_{\mathcal{V}}$ of $E^{\boldsymbol{\gamma}^i}$, $i = \{1, \dots, k\}$ are nested and (2) the parametric problem (2.45) is solvable efficiently by respective algorithms [GGT89; Hoc08; FI03]. In the following, we will show that for a submodular energy E the Joint-DivMBest problem (2.12) reduces to the parametric submodular minimization with the values $\boldsymbol{\gamma}^1 \geq \boldsymbol{\gamma}^2 \geq \dots \geq \boldsymbol{\gamma}^M \in \mathbb{R}^{|\mathcal{V}|}$ given in closed form.

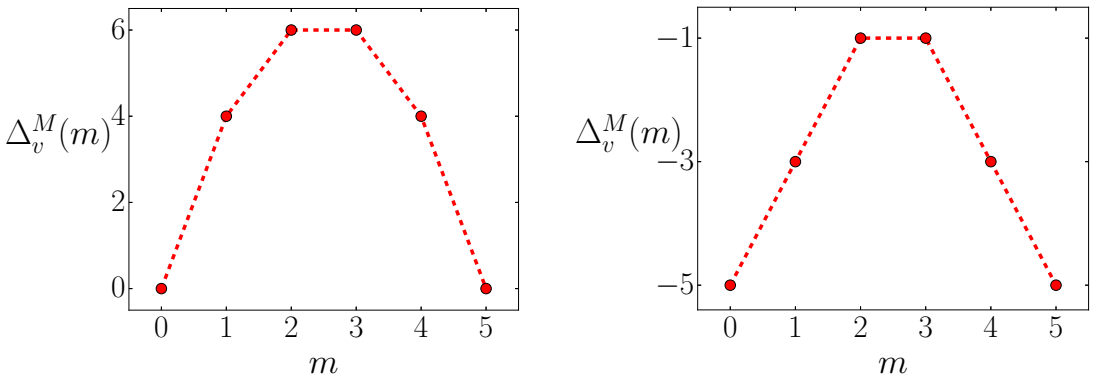


Figure 2.3: Hamming distance (left) and linear (right) diversity measures for $M = 5$. Value m is defined as $\sum_{m=1}^{M} [\![y_v^m = 0]\!]$. Both diversity measures are concave.

Parametric approach for (2.15) Our results hold for the following subclass of the permutation invariant node-wise diversity measures:

Definition 3. *A node-wise diversity measure $\Delta_v^M(m)$ is called* concave *if for any $1 \leq i \leq j \leq M$ it holds*

$$\Delta_v^M(i) - \Delta_v^M(i-1) \geq \Delta_v^M(j) - \Delta_v^M(j-1). \tag{2.46}$$

There are a number of practically relevant concave diversity measures:

Example 2. *Hamming distance diversity* (2.44) *is* concave*, see Fig. 2.3 for illustration.*

Example 3. *Diversity measures of the form*

$$\Delta_v^M(m_v^0) = -\left(|m_v^0 - (M - m_v^0)|\right)^p = -\left(|2m_v^0 - M|\right)^p \tag{2.47}$$

are concave for any $p \geq 1$. Here $M - m_v^0$ is the number of variables labeled as 1*. Hence, $|m_v^0 - (M - m_v^0)|$ is an absolute value of the difference between the numbers of*

variables labeled as 0 *and* 1*. It expresses the natural fact that a distribution of* 0*'s and* 1*'s is more diverse, when their amounts are similar.*

For $p = 1$ we call the measure (2.47) *linear*; for $p = 2$ the measure (2.47) coincides with the Hamming distance diversity (2.44). An illustration of these two cases is given in Fig. 2.3.

Our main theoretical result is given by the following theorem:

Theorem 4. *Let* E *be binary submodular and* Δ^M *be a node-wise diversity measure with each component* Δ_v^M*,* $v \in V$*, being permutation invariant and concave. Then a nested* M*-tuple* $(\boldsymbol{y}^m)_{m=1}^M$ *minimizing the Joint-DivMBest objective* (2.12) *can be found as the solutions of the following* M *problems:*

$$\boldsymbol{y}^m = \arg\min_{\boldsymbol{y}_\mathcal{V}} \left[E(\boldsymbol{y}) + \sum_{v \in \mathcal{V}} \gamma_v^m y_v \right] , \tag{2.48}$$

where $\gamma_v^m = \lambda \left(\Delta_v^M(m) - \Delta_v^M(m-1) \right)$*. In the case of multiple solutions in* (2.48) *the highest minimizer must be selected.*

Proof. We provide the proof of Theorem 4 restricted to pairwise energies. It is based on representing the general multiple diverse solutions problem (2.12) in the form of minimizing a convex multilabel energy. This problem is known as Convex MRF or as total variation (TV) regularized optimization with convex data terms. Thresholding theorems [Hoc01; DS04; CE05; Cha05; Hoc13] then allow to break the problem into independent minimization and connect it to parametric mincut. This approach reveals an important link between our problem and the mentioned methods. It is also the shorter one. However, it is limited by the existing thresholding theorems and does not fully cover *e.g.* the higher order case (as discussed below). We refer to [Kir+16] for the general proof.

For pairwise energies it holds $f = \{u, v\}$, $u, v \in \mathcal{V}$. Therefore, we will denote θ_f as $\theta_{u,v}$. The energy of the master problem (2.8) then reads

$$E(y) = \sum_{v \in \mathcal{V}} \theta_v(y_v) + \sum_{uv \in \mathcal{F}} \theta_{u,v}(y_u, y_v) \, . \tag{2.49}$$

It is known [BVZ01] and straightforward to check that in the binary case it holds

$$E(y) = \text{const} + \sum_{v \in \mathcal{V}} a_v y_v + \sum_{uv \in \mathcal{F}} \Theta_{u,v} |y_u - y_v| \, , \tag{2.50}$$

where $a_v = \theta_v(1) - \theta(0)$ and $\Theta_{u,v} = \theta_{u,v}(0,1) + \theta_{u,v}(1,0) - \theta_{u,v}(0,0) - \theta_{u,v}(1,1)$. For submodular E, the values $\Theta_{u,v}$ are non-negative. In what follows, we will use the representation (2.50) and omit the constant in it, since it does not influence any further considerations.

A nested M-tuple $\{\boldsymbol{y}\}$ is unambiguously specified by $|\mathcal{V}|$ numbers $m_v^0 \in \{0, \dots, M\}$, $v \in \mathcal{V}$, where m_v^0 defines a number of labelings, which are assigned the label 0 in the

node v. The link between the two representations is given by

$$m_v^0 = \sum_m [\![y_v^m = 0]\!], \tag{2.51}$$

$$y_v^m = m \leq m_v^0. \tag{2.52}$$

In other words, labelings y^m are superlevel sets of $m^0 \colon \mathcal{V} \to \{0, \dots, M\}$.

Let us write the general multiple diverse solutions objective (2.15) as a function of m^0. The label $m \in \{0, \dots, M\}$ denotes that exactly m out of M labelings in $\{\boldsymbol{y}\}$ are assigned the label 0 in the node v. The unary cost assigned to a label m in the node v is equal to $a_v(M - m)$, since exactly $(M - m)$ labelings out of M are assigned the label 1 in the node v. The pairwise cost for a pair of labels $\{m, n\}$ in the neighboring nodes $\{u, v\} \in \mathcal{F}$ is equal to $\Theta_{u,v}|m - n|$, since exactly $|m - n|$ labelings switch their label 0 to the label 1 between nodes u and v. Therefore

$$\sum_{i=1}^{M} E(\boldsymbol{y}^i) = \sum_{v \in \mathcal{V}} a_v(M - m_v^0) + \sum_{uv \in \mathcal{F}} \Theta_{u,v} |m_u^0 - m_v^0| \,, \tag{2.53}$$

where m_v^0 is defined as in (2.51).

Adding a node-wise diversity measure $\sum_{v \in \mathcal{V}} \lambda \Delta_v^M(\{\boldsymbol{y}\}_v) = \sum_{v \in \mathcal{V}} \lambda \Delta_v^M(m_v^0)$ and regrouping terms, one obtains that the Joint-DivMBest objective (2.12) is equivalent to

$$\sum_{v \in \mathcal{V}} \big(a_v(M - m_v^0) - \lambda \Delta_v^M(m_v^0) \big) + \sum_{uv \in \mathcal{F}} \Theta_{u,v} |m_u^0 - m_v^0| \tag{2.54}$$

and must be minimized with respect to the labeling $\boldsymbol{m}^0 \in \{0, \dots, M\}^{\mathcal{V}}$.

Since the diversity measure $\lambda \Delta_v^M(m_v^0)$ is concave w.r.t. m_v^0, the unary factors $a_v(M - m_v^0) - \lambda \Delta_v^M(m_v^0)$ are convex. The pairwise factors $\Theta_{u,v}|m_u^0 - m_v^0|$ are also convex w.r.t. $m_u^0 - m_v^0$ due to non-negativity of $\Theta_{u,v}$.

For concave diversity the problem can be solved efficiently in time $O(T(n, m) + n \log M)$ [Hoc01], where $n = |\mathcal{V}|$, $m = |\mathcal{E}|$ and $T(n, m)$ is the complexity of a minimum s-t cut procedure that can be implemented efficiently as parametric. Even for m^0 ranging in the continuous domain the complexity of the method [Hoc01] is polynomial, essentially matching the complexity of a single mincut. In particular, [Hoc01, Theorem 3.1] shows that a solution of such convex multilabel energy minimization problem decouples into M problems of the form (2.48). Our Theorem 4 then follows. □

First note that the sequence $(\boldsymbol{\gamma}^m)_{m=1}^M$ is monotonous due to concavity of Δ_v^M. Each of the M optimization problems (2.48) has the same size as the master problem (2.8) and differs from it by unary potentials only.

Theorem 4 implies that $\boldsymbol{\gamma}^m$ in (2.48) satisfy the monotonicity condition: $\boldsymbol{\gamma}^1 \geq \boldsymbol{\gamma}^2 \geq \ldots \geq \boldsymbol{\gamma}^M$. Therefore, equations (2.48) constitute the parametric submodular minimization problem as defined above, which reduces to the monotonic parametric max-flow problem for pairwise E. Let $\lfloor \cdot \rfloor$ denote the largest integer not exceeding an argument of the operation.

Corollary 3. *Let Δ_v^M in Theorem 4 be the Hamming distance diversity* (2.44)*. Then it holds:*

1. $\gamma_v^m = \lambda(M - 2m + 1)$.

2. The values γ_v^m, $m = 1, \dots, M$ *are symmetrically distributed around* 0*:* $-\gamma_v^m = \gamma_v^{M+1-m} \geq 0$, *for* $m \leq \lfloor (M+1)/2 \rfloor$ *and* $\gamma_v^m = 0$, *if* $m = (M+1)/2$.

3. Moreover, this distribution is uniform, that is $\gamma_v^{m+1} - \gamma_v^m = 2\lambda$, $m = 1, \dots, M$.

4. When M *is odd, the MAP-solution (corresponding to* $\gamma^{(M+1)/2} = 0$*) is* always *among the* M*-best-diverse labelings minimizing* (2.12)*.*

Corollary 4. *Implications 2 and 4 of Corollary 3 hold for any symmetrical concave* Δ_v^M*, i.e. those where* $\Delta_v^M(m) = \Delta_v^M(M + 1 - m)$ *for* $m \leq \lfloor (M+1)/2 \rfloor$.

Corollary 5. *For* linear *diversity measure the value* γ_v^m *in* (2.48) *is equal to* $\lambda \cdot sgn\left(\frac{M}{2} - m\right)$*, where* $sgn(x)$ *is a sign function, i.e.* $sgn(x) = [\![x > 0]\!] - [\![x < 0]\!]$. *Since all* γ_v^m *for* $m < \frac{M}{2}$ *are the same, this diversity measure can give only up to* 3 *different diverse labelings. Therefore, this diversity measure is not useful for* $M > 3$*, and can be seen as a limit of useful concave diversity measures.*

Efficient algorithmic solutions

Theorem 4 suggests several new computational methods for minimizing the general multiple diverse solutions objective (2.15). All of them are more efficient than both ordering based and clique encoding approaches. Indeed, as we show experimentally, they outperform even the sequential DivMBest method (2.3).

The simplest algorithm applies a MAP-inference solver to each of the M problems (2.48) sequentially and independently. This algorithm has the same computational cost as DivMBest (2.3) since it also sequentially solves M problems of the same size. However, already its slightly improved version, described below, performs faster than DivMBest (2.3).

Sequential algorithm. Theorem 4 states that solutions of (2.48) are nested. Therefore, from $y_v^{m-1} = 1$ it follows that $y_v^m = 1$ for labelings $\boldsymbol{y}^{m-1}$ and $\boldsymbol{y}^m$ obtained according to (2.48). This allows to reduce the size and computing time for each subsequent problem in the sequence.[2] Reusing the flow from the previous step gives an additional speedup. In fact, when applying a push relabel or pseudoflow algorithm in this fashion the total work complexity is asymptotically the same as of a single minimum cut [GGT89; Hoc08] of the master problem. In practice, this strategy is efficient with other min-cut solvers (without theoretical guarantees) as well. In our experiments we evaluated it with the dynamic augmenting path method [BK04; KT07].

Parallel algorithm. The M problems (2.48) are completely independent, and their highest minimizers recover the optimal M-tuple $(\mathbf{y^m})_\mathbf{m}$ according to Theorem 4. They can be solved fully in parallel or, using $p < M$ processors, in parallel groups of M/p problems per processor, incrementally within each group. The overhead is only in copying data costs and sharing the memory bandwidth.

[2] By applying "symmetric reasoning" for the label 0, further speed-ups can be achieved. However, we stick to the first variant in our experiments.

Alternative approaches One may suggest that for large M it would be more efficient to solve the full parametric maxflow problem [Hoc08; GGT89] and then "read out" solutions corresponding to the desired values γ^m. However, the known algorithms [Hoc08; GGT89] would perform exactly the incremental computation described in the sequential approach above plus an extra work of identifying all breakpoints. This is only sensible when M is larger than the number of breakpoints or the diversity measure is not known in advance (e.g. is itself parametric). Similarly, parametric submodular function minimization can be solved in the same worst case complexity [FI03] as non-parametric, but the algorithm is again incremental and would just perform less work when the parameters of interest are known in advance.

2.6 Experimental Evaluation

We base our experiments on three datasets: (a) interactive foreground/background segmentation for images with provided scribbles annotations [4], (b) multiclass semantic segmentation on Pascal VOC 2012 [Eve+15], and (c) a new foreground/background segmentation dataset derived from Pascal 2012 [Eve+15].

Baselines. Our main competitor is the fastest known approach for inferring M diverse solutions, greedy optimization of (2.12), the `DivMBest` method [Bat+12]. We made its efficient re-implementation using dynamic graph-cut [KT07].

Diversity Measure. In our work we present methods that deal with node-wise diversity measures (2.9) only. We use the Hamming distance diversity measure (2.11) in all of experimental evaluation. Note that in [PJB14] more sophisticated diversity measures were used e.g. the *Hamming Ball*. However, the `DivMBest` method (2.3) with this measure requires to run a very time-consuming *HOP-MAP* [TGZ10] inference technique. Moreover, the experimental evaluation in [Kir+15b] suggests that global minimization of (2.12) with Hamming distance diversity (2.11) outperforms `DivMBest` with a *Hamming Ball* distance diversity.

Our methods. In our thesis we present three types of global optimization techniques for (2.12) that apply to different types of original energy and diversity measures:

- Clique Encoding (denoted as `CE`) and K-truncated Clique Encoding (denoted as $\texttt{CE}_K$) methods that are applicable to any solvable pair-wise original energy and a node-wise diversity measure (2.9).
- Ordering based method that solves the problem (2.34) with the Hamming diversity measure (2.11) by transforming it into min-cut/max-flow problem [KZ04; SF06; Ish03] and running the solver [BK04] is denoted as `Ordering-Global`. The method is applicable to any submodular original energy and node-wise diversity measures that satisfy constraints of Lemma 2.
- Parametric based methods described in Section 2.5.4, i.e. sequential and parallel. We refer to them as `Parametric-sequential` and `Parametric-parallel` respectively. We utilize the dynamic graph-cut [KT07] technique for

`Parametric-sequential`, which makes it comparable to our implementation of `DivMBest`. The max-flow solver of [BK04] is used for `Parametric-parallel` together with *OpenMP* directives. For the experiments we use a computer with 6 physical cores (12 virtual cores), and run `Parametric-parallel` with M threads. Parametric methods are applicable to any binary submodular original energy and diversity measures that satisfy constraints of the Theorem 4.

In our experimental evaluation we compare these methods to the greedy approach and also compare them with each other.

Parameters λ from (2.12) were tuned via cross-validation for each algorithm and each experiment separately.

2.6.1 Datasets

Interactive segmentation. Instead of returning a single segmentation corresponding to a MAP-solution, diversity methods return a small number of possible low-energy results. Following [Bat+12] we model only the first iteration of such an interactive procedure, i.e. we consider user scribbles to be given and compare the sets of segmentations returned by the compared diversity methods.

Authors of [Bat+12] kindly provided us with their 50 graphical model instances, corresponding to the MAP-inference problem (2.8). They are based on a subset of the PASCAL VOC 2010 [Eve+15] segmentation challenge with manually added scribbles. Pairwise potentials constitute contrast sensitive Potts terms [BJ01] which implies that the MAP-inference is submodular and therefore is solvable by min-cut/max-flow algorithms [KZ04].

Semantic segmentation. The category level segmentation from PASCAL VOC 2012 challenge [Eve+15] contains 1449 validation images with known ground truth which we used for evaluation of diversity methods. Corresponding pairwise models with contrast sensitive Potts terms of the form $\theta_{uv}(y, y') = w_{uv}[\![y \neq y']\!]$, $uv \in \mathcal{F}$, were used in [PJB14] and kindly provided to us by the authors. Contrary to interactive segmentation, the label sets contain 21 elements and hence the respective MAP-inference problem (2.8) is not submodular anymore. However it still can be approximatively solved by α-expansion or α-β-swap.

Foreground/background segmentation. The Pascal VOC 2012 [Eve+15] segmentation dataset has 21 labels. We selected all those 451 images from the validation set for which the ground truth labeling has only two labels (background and one of the 20 object classes) and which were *not* used for training. As unary potentials we use the output probabilities of the publicly available fully convolutional neural network `FCN-8s` [LSD15] which is trained for the Pascal VOC 2012 challenge. This CNN gives unary terms for all 21 classes. For each image we pick only two classes: the background and the class-label that is presented in the ground truth. As pairwise potentials we use the contrastive-sensitive Potts terms [BJ01] with a 4-connected grid structure. Resulting energy is submodular. We use this new dataset to evaluate performance of Parametric-based method.

	M=2		M=6		M=10	
	quality	time	quality	time	quality	time
`DivMBest` [Bat+12]	93.16	2.6	95.02	11.6	95.16	15.4
`CE`	**95.13**	6.8	**96.01**	74.3	**96.19**	1247
`Ordering-Global`	**95.13**	5.5	**96.01**	17.2	**96.19**	80.3
`Parametric-sequential` (1 core)	**95.13**	2.2	**96.01**	5.5	**96.19**	8.4
`Parametric-parallel` (6 cores)	**95.13**	**1.9**	**96.01**	**4.3**	**96.19**	**6.2**

Table 2.1: **Interactive segmentation**. The quality measure is a per-pixel accuracy of the best segmentation, out of M, averaged over all test images. The runtime is in milliseconds (ms). The quality for $M = 1$ is 91.57. `Parametric-parallel` is the fastest method followed by `Parametric-sequential`. Both achieve higher quality than `DivMBest`, and return the same solution as `Ordering-Global` and `CE`.

2.6.2 Clique Encoding

Clique encoding (`CE`) method is applicable to pairwise energies. In our experiments we used α-expansion [BVZ01], which turns into the max-flow algorithm in case of two labels. Table 2.1 shows its comparison with DivMBest-based techniques for the **interactive segmentation** dataset. As quality measure we used per pixel accuracy of the best solution for each sample averaged over all test images. Parameter λ has been chosen for each method separately via cross-validation. In all these experiments, our `CE` method shows significantly better accuracy than its competitors. Fig. 2.4 shows several examples of clique encoding output and its comparison with DivMBest. Running time of our `CE` method is, as expected, higher than those for `DivMBest`, however it still can be considered as practically useful.

Pascal VOC **multiclass semantic segmentation** dataset has 21 labels. Because of a significant number of labels we were unable to use `CE` approach for $M > 5$ and resorted to $\texttt{CE}_3$. Results of the quantitative evaluation are presented in Table 2.2, where each method was used with parameter λ optimally tuned via cross-validation on validation set in PASCAL VOC 2012. Following [Bat+12], as quality measure we used Intersection-over-Union (IoU) of the best solution for each sample averaged over all test images. Exemplary comparison of `CE` and `DivMBest` is shown in Fig. 2.5. It turns out that even the suboptimal optimization method $\texttt{CE}_3$ outperforms *all* competitors, except `CE`, which show even better segmentation accuracy. The methods $\texttt{CE}_3$ is a hybrid of `DivMBest` and `CE` delivering a reasonable trade-off between running time and accuracy of inference for the model E^M (2.12).

2.6.3 Ordering Based

Interactive segmentation datasets has binary submodular energies, therefore, Theorem 3 is applicable and exact solution of general diversity problem (2.38) can be found by solving single submodular minimization (2.37). In our experiments we transform this minimization into min-cut/max-flow problem [KZ04; SF06; Ish03] and running the solver [BK04]. This approach is denoted as `Ordering-Global`.

Quantitative comparison and run-time of the considered methods are provided

Figure 2.4: Comparison for samples from interactive segmentation dataset. Number above each solution is a corresponding per pixel accuracy.

in Table 2.1, where each method was used with the parameter λ (see (2.3), (2.12)), optimally tuned via cross-validation. Following [Bat+12], as a quality measure we used the per pixel accuracy of the best solution for each sample averaged over all test images. Methods `CE` and `Ordering-Global` gave the same quality, which confirms the observation made in [Kir+15a], that `CE` returns an exact MAP solution for each sample in this dataset. The run-time provided is also averaged over all samples. The max-flow algorithm was used for `DivMBest` and `Ordering-Global` and α-expansion for `CE`.

It can be seen that the `Ordering-Global` qualitatively outperforms `DivMBest` and is equal to `CE`. However, it is considerably faster than the latter (the difference grows exponentially with M) and the runtime is of the same order of magnitude as the one of `DivMBest`.

In our experiments with Pascal VOC 2012 **multiclass semantic segmentation** dataset, we use energies with contrast sensitive Potts terms which are non-submodular in a multilabel case. Since the MAP-inference problem (2.8) is not submodular in this experiment, Theorem 3 is not applicable. We used two ways to overcome it. *First*, we modified the diversity potentials according to (2.38), as if Theorem 3 were to be correct. This basically means we were explicitly looking for ordered M best diverse

	MAP inference	M=5		M=15		M=16	
		quality	time	quality	time	quality	time
DivMBest	α-exp[BJ01]	51.21	**0.01**	52.90	**0.03**	53.07	**0.03**
CE	α-exp[BJ01]	**54.22**	733	-	-	-	-
CE_3	α-exp[BJ01]	54.14	2.28	**57.76**	5.87	**58.36**	7.24
Ordering-Global-forced	α-β-swap[BJ01]	53.81	0.01	56.08	0.08	56.31	0.08
Ordering-Global-learned	max-flow[BK04]	53.85	0.38	56.14	35.47	56.33	38.67
Ordering-Global-learned	α-exp[BJ01]	53.84	0.01	56.08	0.08	56.31	0.08

Table 2.2: PASCAL VOC 2012 multiclass semantic segmentation. Intersection over union quality measure/running time. The best segmentation out of M is considered. Compare to the average quality 43.51 of a single labeling. Time is in seconds (s). Notation '-' correspond to absence of result due to computational reasons or inapplicability of the method. (*)- methods were not run by us and the results were taken from [PJB14] directly. The MAP-inference column references the slowest inference technique out of those used by the method.

labelings. The resulting inference problem was addressed with α-β-swap (since neither max-flow nor the α-expansion algorithms are applicable). We refer to this method as to Ordering-Global-forced. *The second* way to overcome the non-submodularity problem is based on learning. Using structured SVM technique we trained pairwise potentials with additional constraints enforcing their submodularity, as it is done in e.g. [FS08]. We kept the contrast terms w_{uv} and learned only a single submodular function $\hat{\theta}(y, y')$, which we used in place of $[\![y \neq y']\!]$. After the learning all our potentials had the form $\theta_{uv}(y, y') = w_{uv}\hat{\theta}(y, y')$, $uv \in \mathcal{F}$. We refer to this method as to Ordering-Global-learned. For the model we use max-flow[BK04] as an exact inference method and α-expansion[BJ01] as a fast approximate inference method.

Quantitative comparison and run-time of the considered methods is provided in Table 2.2, where each method was used with the parameter λ (see (2.3), (2.12)) optimally tuned via cross-validation on the validation set in PASCAL VOC 2012. Following [Bat+12], we used the Intersection over union quality measure, averaged over all images. Among combined methods with higher order diversity measures we selected only those providing the best results. Quantitative results delivered by Ordering-Global-foced and Ordering-Global-learned are very similar (though the latter is negligibly better), significantly outperform those of DivMBest and are only slightly inferior to those of CE_3. However the run-time for Ordering-Global-forced and α-expansion version of Ordering-Global-learned are comparable to those of DivMBest and outperform all other competitors due to the use of the fast inference algorithms and linearly growing label space, contrary to the label space of CE_3, which grows as $(L_v)^3$.

2.6.4 Parametric Based

Parametric based method is applicable to binary submodular original energies and permutation-invariant concave diversity measures. Energies from **interactive segmentation** dataset and Hamming distance satisfy these constraints. Quantitative comparison

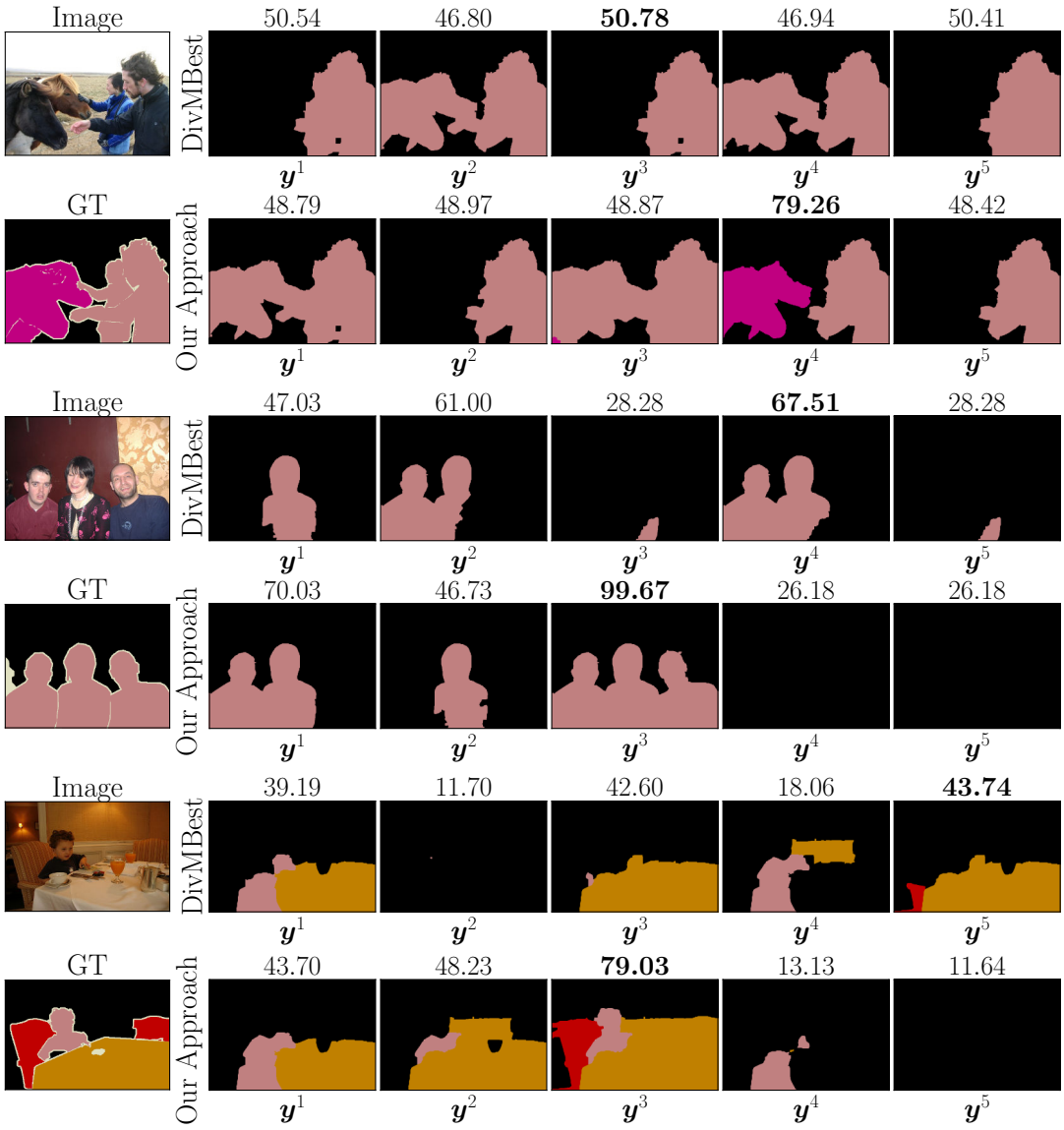
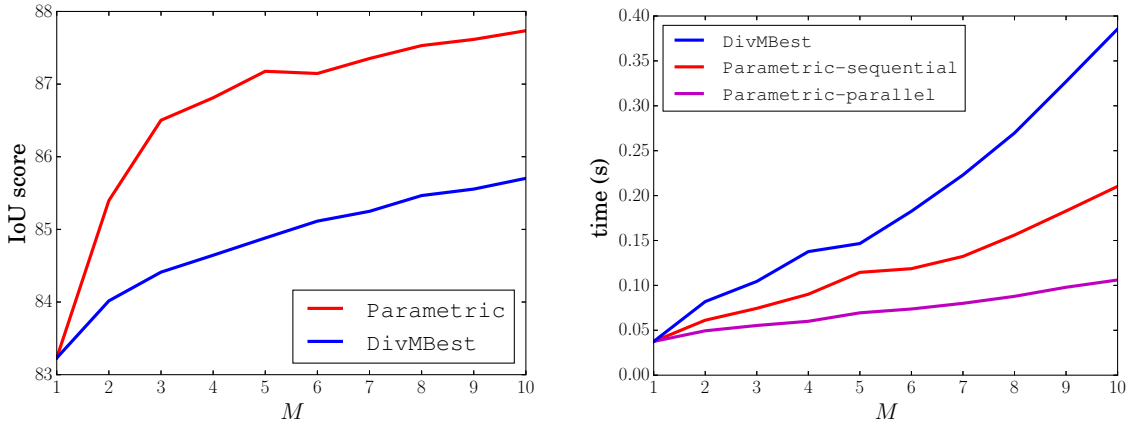


Figure 2.5: Comparison for samples from Pascal VOC 2012 dataset. Number above each solution is a corresponding intersection over union quality measure.

and runtime of the different algorithms are presented in Table 2.1. As in [Bat+12], our quality measure is an Intersection-over-Union (IoU) of the best solution for each test image, averaged over all test images. As expected, `Ordering-Global` and `Parametric-*` return the same, exact solution of (2.12). The measured runtime is also averaged over all test images. `Parametric-parallel` is the fastest method followed by `Parametric-sequential`. Note that on a computer with fewer cores, `Parametric-sequential` may even outperform `Parametric-parallel` because of the parallelization overheads.

New **foreground/background** dataset has binary submodular energies too. As quality measure we use the standard Pascal VOC measure for semantic segmentation – average intersection-over-union (IoU) [Eve+15]. The unary potentials alone, i.e. output of `FCN-8s`, give 82.12 IoU. The single best labeling, returned by the MAP-inference problem, improves it to 83.23 IoU.



(a) Intersection-over-union (IoU)

(b) Runtime in seconds

Figure 2.6: **Foreground/background segmentation**. (a) Intersection-over-union (IoU) score for the best segmentation, out of M. `Parametric` represents a curve, which is the same for `Parametric-sequential`, `Parametric-parallel` and `Ordering-Global`, since they exactly solve the same Ordering-Global problem. (b) `DivMBest` uses dynamic graph-cut [KT07]. `Parametric-sequential` uses dynamic graph-cut and a reduced size graph for each consecutive labeling problem. `Parametric-parallel` solves M problems in parallel using `OpenMP`.

The comparisons with respect to runtime and accuracy of results are presented in Fig. 2.6a and 2.6b respectively. The increase in runtime with respect to M for `Parametric-parallel` is due to parallelization overhead costs, which grow with M. `Parametric-parallel` is a clear winner in this experiment, both in terms of quality and runtime. `Parametric-sequential` is slower than `Parametric-parallel` but faster than `DivMBest`. The difference in runtime between these three algorithms grows with M.

2.7 Conclusion

In this chapter we explore global diversity optimization problem that produces multiple diverse solutions for a single trained model. We show that other techniques generating diverse solutions can be seen as special cases for the new problem formulation. We present several optimization approximate and exact optimization techniques for the new optimization objective that have different requirements to original model. Our work presents a practical guide for figuring out the right optimization strategy for a given problem with its constraints. We hope that this guide will help to handle ambiguity in real-world applications and will facilitate further research in this direction.

Chapter 3

Bottom-Up Approach for Instance Segmentation

3.1 Introduction

This chapter addresses the task of segmenting each individual instance of a semantic class in an image. The task is known as *instance-aware semantic segmentation*, in short *instance segmentation*, and is a more refined task than semantic segmentation, where each pixel is only labeled with its semantic class. An example of semantic segmentation and instance segmentation is shown in Fig. 3.1a-3.1b. While semantic segmentation has been a very popular problem to work on in the last half decade, the interest in instance segmentation has significantly increased recently. This is not surprising since semantic segmentation has already reached a high level of accuracy, in contrast to the harder task of instance segmentation. Also, from an application perspective there are many systems, such as autonomous driving or robotics, where a more detailed understanding of the surrounding is important for acting correctly in the world.

In recent years, Convolutional Neural Networks (CNN) have tremendously increased the performance of many computer vision tasks. This is also true for the task of instance segmentation, see the benchmarks [Cor+16; Lin+14]. However, for this task it is, in our view, not clear whether the best modelling-paradigm has already been found. Hence, the motivation of this work is to explore a new, and very different, modelling-paradigm. To be more precise, we believe that the problem of instance segmentation has four core challenges, which any method has to address. Firstly, the label of an instance, e.g. "car number 5", does not have a meaning, in contrast to semantic segmentation, e.g. class "cars". Secondly, the number of instances in an image can vary greatly, e.g. between 0 and 120 for an image in the CityScapes dataset [Cor+16]. Thirdly, in contrast to object detection with bounding boxes, each instance (a bounding box) cannot simply be described by four numbers (corners of bounding box), but has to be described by a set of pixels. Finally, in contrast to semantic segmentation, a more refined labeling of the training data is needed, i.e. each instance has to be segmented separately. Especially for rare classes, e.g. motorcycles, the amount of training data, which is available nowadays, may not be sufficient. Despite these challenges, the state of the art techniques for instance segmentation are CNN-based. As an example, [DHS16; Zag+16] address these challenges with a complex multi-loss cascade CNN architectures, which are,

(a) (b)

(c) (d)

Figure 3.1: An image from the CityScapes dataset [Cor+16]: (a) Ground truth semantic segmentation, where all cars have the same label. (b) The ground truth instance segmentation, where each instance, i.e. object, is highlighted by a distinct color. In this chapter we use a "limiting" definition of instance segmentation, in the sense that each instance must be a connected component. Despite this limitation, we will demonstrate high-quality results. (c) Shows the result of our InstanceCut method. As can be seen, the front car is split into two instances, in contrast to (b). (d) Our connected-component instances are defined via two output modalities: (i) the semantic segmentation, (ii) all instance boundaries (shown in bold-black).

however, difficult to train. In contrast, our modelling-paradigm is very different to standard CNN-based architectures: assume that each pixel is assigned to one semantic class, and additionally we insert some edges (in-between pixels) which form loops – then we have solved the problem of instance segmentation! Each connected region, enclosed by a loop of instance-aware edges is an individual instance where the class labels of the interior pixels define its class. These are exactly the ingredients of our approach: (i) a standard CNN that outputs an instance-agnostic semantic segmentation, and (ii) a new CNN that outputs all boundaries of instances. In order to make sure that instance-boundaries encircle a connected component, and that the interior of a component has the same class label, we combine these two outputs into a novel multi-cut formulation. We call our approach *InstanceCut*.

Our InstanceCut approach has some advantages and disadvantages, which we discuss next. With respect to this, we would like to stress that these pros and cons are, however, quite different to existing approaches. This means that in the future we envision that our approach may play an important role, as a subcomponent in an "ultimate" instance segmentation system. Let us first consider the limitations, and then the advantages. The minor limitation of our approach is that, obviously, we cannot find instances that are formed by disconnected regions in the image (see Fig. 3.1b-3.1c). However, despite this limitation, our method demonstrates promising results in terms of accuracy. In the future, we foresee various ways to overcome this limitation, e.g. by reasoning globally about shape.

We see the following major advantages of our approach. Firstly, all the four

challenges for instance segmentation methods, listed above, are addressed in an elegant way: (i) the multi-cut formulation does not need a unique label for an instance; (ii) the number of instances arises naturally from the solution of the multi-cut; (iii) our formulation is on the pixel (superpixel) level; (iv) since we do not train a CNN for segmenting instances globally, our approach deals very well with instances of rare classes, as they do not need special treatment. Finally, our InstanceCut approach has another major advantage, from a practical perspective. We can employ any semantic segmentation method, as long as it provides pixel-wise log-probabilities for each class. Therefore, advances in this field may directly translate to an improvement of our method. Also, semantic segmentation, here a Fully-Convolutional-Neural-Network (FCN) [YK16], is part of our new edge-detection approach. Again, advances in semantic segmentation may improve the performance of this component, as well.

Our Contributions in short form are:

- We propose a novel paradigm for instance-aware semantic segmentation, which has different pros and cons than existing approaches. In our approach, we only train classifiers for semantic segmentation and instance-edge detection, and not directly any classifier for dealing with global properties of an instance, such as shape.
- We propose a novel MultiCut formulation that reasons globally about the optimal partitioning of an image into instances.
- We propose a new FCN-based architecture for instance-aware edge detection.
- We validate experimentally that our approach achieves strong result, and performs particularly well for rare object classes.

3.2 Related Work

Proposal-based methods. This group of methods uses detection or a proposal generation mechanism as a subroutine in the instance-aware segmentation pipeline.

Several recent methods decompose the instance-aware segmentation problem into a detection stage and a foreground/background segmentation stage [DHS16; Har+15]. These methods propose an end-to-end training that incorporates all parts of the model. In addition, non-maximal suppression (NMS) may be employed as a post-processing step. A very similar approach generates proposals using e.g. MCG [Arb+14] and then, in the second stage, a different network classifies these proposals [Cor+16; Har+14; DHS15; CLY15].

Several methods produce proposals for instance segmentations and combine them, based on learned scores [Lia+16; PCD15; Pin+16] or generate parts of instances and then combine them [Dai+16; Liu+16].

Although the proposal-based methods show state-of-the-art performance on important challenges, Pascal VOC2012 [Eve+15] and MSCOCO [Lin+14], they are limited by the quality of the used detector or proposal generator. Our method is, in turn, dependent on the quality of the used semantic segmentation. However, for the latter a considerable amount of research exists with high quality results.

Proposal-free methods. Recently, a number of alternative techniques to proposal-based approaches have been suggested in the literature. These methods explore different decompositions of instance-aware semantic segmentation followed by a post-processing step that assembles results.

In [Uhr+16] the authors propose a template matching scheme for instance-aware segmentation based on three modalities: predicted semantic segmentation, depth estimation, and per-pixel direction estimation with respect to the center of the corresponding instance. The approach requires depth data for training and does not perform well on highly occluded objects.

Another work, which focuses on instance segmentation of cars [Zha+15; ZFU16] employs a conditional random field that reasons about instances using multiple overlapping outputs of an FCN. The latter predicts a fixed number of instances and their order within the receptive field of the FCN, i.e. for each pixel, the FCN predicts an ID of the corresponding instance or background label. However, in these methods the maximal number of instances per image must be fixed in advance. A very large number may have a negative influence on the system performances. Therefore, this method may not be well-suited for the CityScapes dataset, where the number of instances varies considerably among images.

In [WSH16] the authors predict the bounding box of an instance for each pixel, based on instance-agnostic semantic segmentation. A post-processing step filters out the resulting instances. Recurrent approaches produce instances one-by-one. In [RZ16] an attention-based recurrent neural network is presented. In [RPT16] an LSTM-based [HS97] approach is proposed. The work [Lia+17] presents a proposal-free network that produces an instance-agnostic semantic segmentation, number of instances for the image, and a per-pixel bounding box of the corresponding instance. The resulting instance segmentation is obtained by clustering. The method is highly sensitive to the right prediction of the number of instances. We also present a proposal-free method. However, ours is very different in paradigm. To infer instances, it combines semantic segmentation and object boundary detection via global reasoning.

3.3 InstanceCut

3.3.1 Overview of the proposed framework

We begin with presenting a general pipeline of our new InstanceCut framework (see Fig. 3.2) and then describe each component in detail. The first two blocks of the pipeline are processed independently: *semantic segmentation* and *instance-aware edge detection* operate directly on the input image. The third, *image partitioning block*, reasons about instance segmentation on the basis of the output provided by the two blocks above.

More formally, the semantic segmentation block (Section 3.3.2) outputs a log-probability of a semantic class $a_{i,l}$ for each class label $l \in \mathcal{L} = \{0, 1 \dots, L\}$ and each pixel i of the input image. We call $a_{i,l}$, *per-pixel semantic class scores*. Labels $1, \dots, L$ correspond to different semantic classes and 0 stands for background.

Independently, the instance-aware edge detection (Section 3.3.3) outputs log-probabilities b_i of an object boundary for each pixel i. In other words, b_i indicates how likely it is that pixel i touches an object boundary. We term b_i as *a per-pixel instance-aware edge*

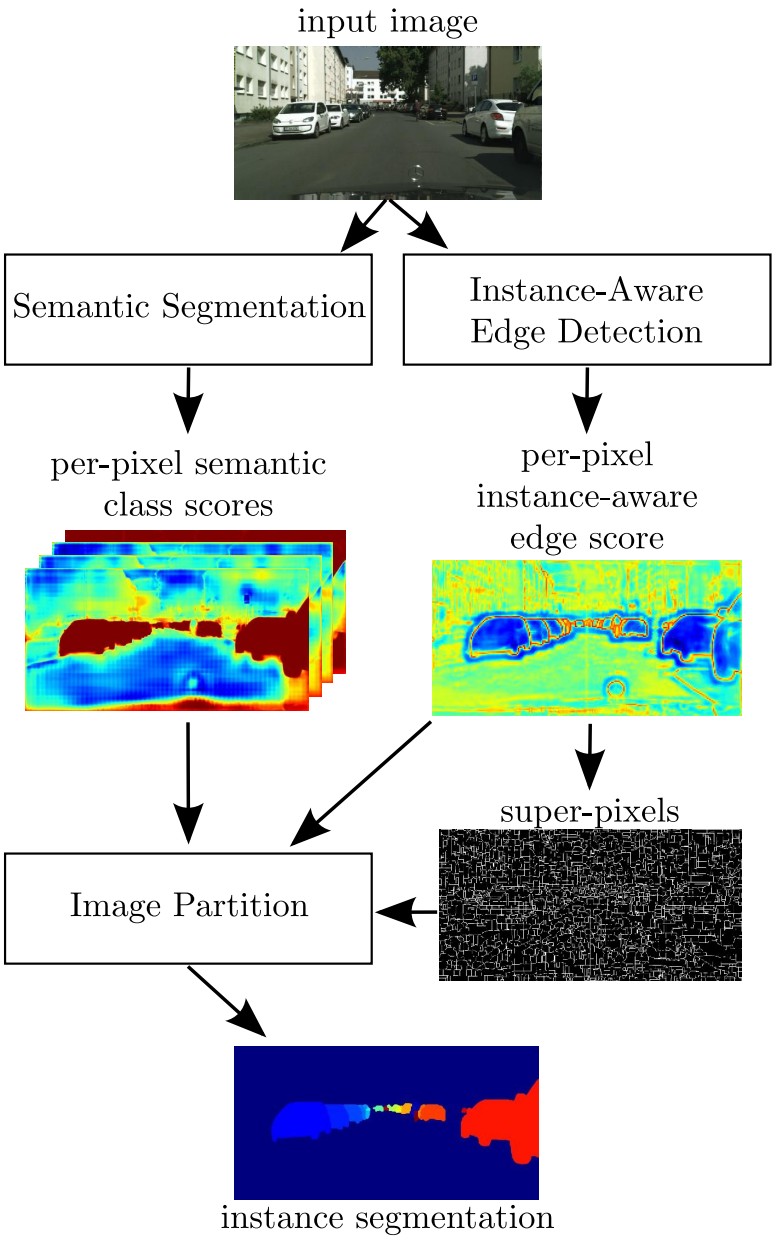


Figure 3.2: **Our InstanceCut pipeline - Overview.** Given an input image, two independent branches produce the per-pixel semantic class scores and per-pixel instance-aware edge scores. The edge scores are used to extract superpixels. The final image partitioning block merges the superpixels into connected components with a class label assigned to each component. The resulting components correspond to object instances and background.

score. Note that these scores are class-agnostic.

Finally, the image partitioning block outputs the resulting instance segmentation, obtained using the semantic class scores and the instance-aware edge scores. We refer to Section 3.3.4 for a description of the corresponding optimization problem. To speed-up optimization, we reduce the problem size by resorting to a superpixel image. For the superpixel extraction we utilize the well-known watershed technique [VS91], which is run directly on the edge scores. This approach efficiently ensures that the extracted

superpixel boundaries are aligned with boundaries of the instance-aware edge scores.

3.3.2 Semantic Segmentation

Recently proposed semantic segmentation frameworks are mainly based on the fully convolution network (FCN) architecture. Since the work [LSD15], many new FCN architectures were proposed for this task [YK16; GF16]. Some of the methods utilize a conditional random field (CRF) model on top of an FCN [Che+17a; LSR+16], or incorporate CRF-based mechanisms directly into a network architecture [Liu+15; Zhe+15; SU15]. Current state-of-the-art methods report around 78% mean Intersection-over-Union (IoU) for the CityScapes dataset [Cor+16] and about 82% for the PASCAL VOC2012 challenge [Eve+15]. Due to the recent progress in this field, one may say that with a sufficiently large dataset, with associated dense ground truth annotation, an FCN is able to predict semantic class for each pixel with high accuracy.

In our experiments, we employ two publicly available pre-trained FCNs: Dilation10 [YK16] and LRR-4x [GF16]. These networks have been trained by the respective authors and we can also use them as provided, without any fine-tuning. Note, that we also use the CNN-CRF frameworks [Zhe+15; Che+17a] with dense CRF [Kol11], since dense CRF's output can also be treated as the log-probability scores $a_{i,l}$.

Since our image partitioning framework works on the superpixel level we transform the pixel-wise semantic class scores $a_{i,l}$ to the superpixel-wise ones $a_{u,l}$ (here u indexes the superpixels) by averaging the corresponding pixels' scores.

3.3.3 Instance-Aware Edge Detection

Let us first review existing work, before we describe our approach. Edge detection (also know as *boundary detection*) is a very well studied problem in computer vision. The classical results were obtained already back in the 80's [Can86]. More recent methods are based on spectral clustering [SM00; Arb+11; Arb+14; Iso+14]. These methods perform global inference on the whole image. An alternative approach suggests to treat the problem as a per-pixel classification task [LZD13; DZ15]. Recent advances in deep learning have made this class of methods especially efficient, since they automatically obtain rich feature representation for classification [GL14; Kiv+14; She+15; BST15a; BST15b; XT15; BST16].

The recent per-pixel classification method [BST16] constructs features, which are based on an FCN trained for semantic segmentation on Pascal VOC 2012 [Eve+15]. The method produces state-of-the-art edge detection performance on the BSD500 dataset [Arb+11]. The features for each pixel are designed as a concatenation of intermediate FCN features, corresponding to that particular pixel. The logistic regression trained on these features, followed by non-maximal suppression, outputs a per-pixel edge probability map. The paper suggests that the intermediate features of an FCN trained for semantic segmentation form a strong signal for solving the edge detection problem. Similarly constructed features also have been used successfully for other dense labelling problems [Har+15].

For datasets like BSDS500 [Arb+11] most works consider general edge detection problem, where annotated edges are class- and instance-agnostic contours. In our work the instance-aware edge detection outputs a probability for each pixel, whether it

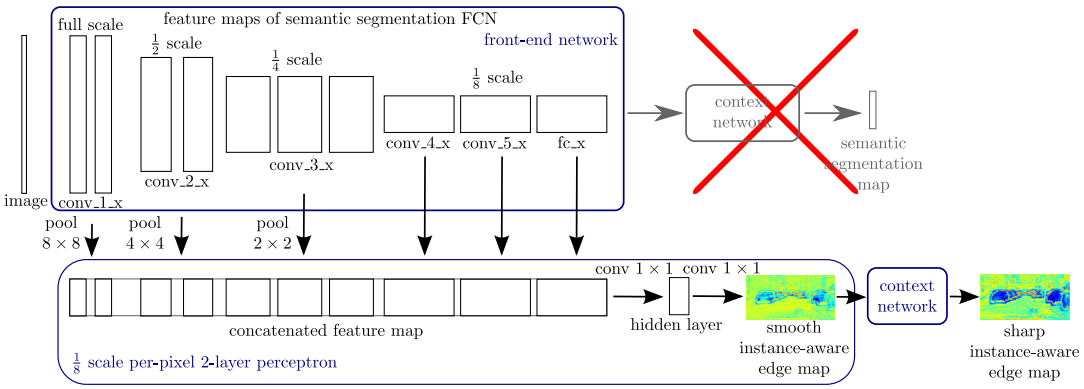


Figure 3.3: **Instance-aware edge detection block.** The semantic segmentation FCN is the front-end part of the network [YK16] trained for semantic segmentation on the same dataset. Its intermediate feature maps are downsampled, according to the size of the smallest feature map, by a max-pooling operation with an appropriate stride. The concatenation of the downsampled maps is used as a feature representation for a per-pixel 2-layer perceptron. The output of the perceptron is refined by a context network of Dilation10 [YK16] architecture.

touches a boundary. This problem is more challenging than canonical edge detection, since it requires to reason about contours and semantics jointly, distinguishing the true objects' boundaries and other not relevant edges, e.g. inside the object or in the background. Below (see Fig. 3.3), we describe a new network architecture for this task that utilizes the idea of the intermediate FCN features concatenation.

As a base for our network we use an FCN that is trained for semantic segmentation on the dataset that we want to use for object boundary prediction. In our experiments we use a pre-trained Dilation10 [YK16] model, however, our approach is not limited to this architecture and can utilize any other FCN-like architectures. We form a per-pixel feature representation by concatenating the intermediate feature maps of the semantic segmentation network. This is based on the following intuition: during inference, the semantic segmentation network is able to identify positions of transitions between semantic classes in the image. Therefore, its intermediate features are likely to contain a signal that helps to find the borders between classes. We believe that the same features can be useful to determine boundaries between objects.

Commonly used approaches [BST16; Har+15] suggest upscaling feature maps that have a size which is smaller than the original image to get per-pixel representation. However, in our experiments such an approach produces thick and over-smooth edge scores. This behavior can be explained by the fact that the most informative feature maps have an 8 times smaller scale than the original image. Hence, instead of upscaling, we downscale all feature maps to the size of the smallest map. Since the network was trained with rectified linear unit (ReLU) activations, the active neurons tends to output large values, therefore, we use max-pooling with a proper stride for the downscaling, see Fig. 3.3.

The procedure outputs the downscaled feature maps (of a *semantic segmentation FCN*, see Fig. 3.3) that are *concatenated* to get the downscaled per-pixel *feature map*. We utilize a *2-layer perceptron* that takes this feature map as input and outputs log-probabilities for edges (*smooth instance-aware edge map*, see Fig. 3.3). The perceptron

method is the same for all spatial positions, therefore, it can be represented as two layers of 1×1 convolutions with the ReLU activation in between.

In our experiments we have noticed that the FCN gives smooth edge scores. Therefore, we apply *a context network* [YK16] that refines the scores making them sharper. The new architecture is an FCN, i.e. it can be applied to images of arbitrary size, it is differentiable and has a single loss at the end. Hence, straightforward end-to-end training can be applied for the new architecture. We upscale the resulting output map to match an input image size.

Since the image partition framework, that comes next, operates on super-pixels, we need to transform the per-pixel edge scores b_i to edge scores $b_{u,v}$ for each pair $\{u, v\}$ of neighboring superpixels. We do this by averaging all scores of of those pixels that touch the border between u and v.

In the following, we describe an efficient implementation of the 2-layer perceptron and also discuss our training data for the boundary detection problem.

Efficient implementation. In our experiments, the input for the 2-layer perceptron contains about 13k features per pixel. Therefore, the first layer of the perceptron consumes a lot of memory. It is, however, possible to avoid this by using a more efficient implementation. Indeed, the first layer of the perceptron is equivalent to the summation of outputs of multiple 1×1 convolutions, which are applied to each feature map independently. For example, `conv_1` is applied to the feature maps from the `conv_1_x` intermediate layer, `conv_2` is applied to the feature maps from `conv_2_x` and its output is summed up with the output of `conv_1`, etc. This approach allows reducing the memory consumption, since the convolutions can be applied during evaluation of the front-end network.

Training data. Although it is common for ground truth data that object boundaries lie in-between pixels, we will use in the following the notion that a boundary lies on a pixel. Namely, we will assume that a pixel i is labeled as a boundary if there is a neighboring pixel j, which is assigned to a different object (or background). Given the size of modern images, this boundary extrapolation does not affect performance. As a ground truth for boundary detection we use the boundaries of object instances presented in CityScapes [Cor+16].

As mentioned in several previous works [XT15; BST15b], highly unbalanced ground truth (GT) data heavily harms the learning progress. For example, in BSDS500 [Arb+11] less than 10% of pixels on average are labeled as edges. Our ground truth data is even more unbalanced: since we consider the object boundaries only, less than 1% of pixels are labeled as being an edge. We employ two techniques to overcome this problem of training with unbalanced data: *a balanced loss function* [XT15; HL15] and *pruning of the ground truth data*.

The balanced loss function [XT15; HL15] adds a coefficient to the standard log-likelihood loss that decreases the influence of errors with respect to classes that have a lot of training data. That is, for each pixel i the balanced loss is defined as

$$\begin{aligned} loss(p_{edge}, y^{GT}) = &[\![y^{GT} = 1]\!] \log(p_{edge}) \\ &+ \alpha [\![y^{GT} = 0]\!] \log(1 - p_{edge}) \,, \end{aligned} \tag{3.1}$$

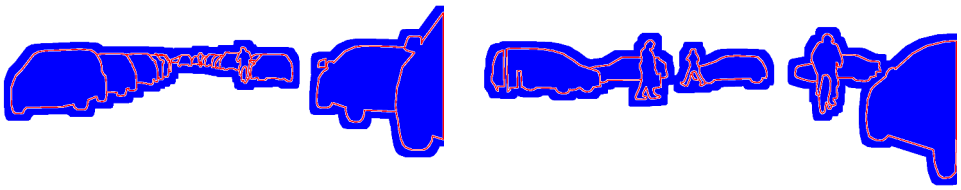


Figure 3.4: Ground truth examples for our instance-aware edge detector. Red indicates pixels that are labeled as edges, blue indicates background, i.e. no edge and white pixels are ignore.

where $p_{edge} = 1/(1 - e^{-b_i})$ is the probability of the pixel i to be labeled as an edge, y^{GT} is the ground truth label for i (the label 1 corresponds to an edge), and $\alpha = N_1/N_0$ is the balancing coefficient. Here, N_1 and N_0 are numbers of pixels labeled, respectively, as 1 and 0 in the ground truth.

Another way to decrease the effect of unbalanced GT data is to subsample the GT pixels, see e.g. [BST16]. Since we are interested in instance-aware edge detection and combine its output with our semantic segmentation framework, a wrong edge detection, which is far from the target objects (for example, in the sky) does not harm the overall performance of the InstanceCut framework. Hence, we consider a pixel to be labeled as background for the instance-aware edge detection if and only if it lies inside the target objects, or in an area close to it, see Fig. 3.4 for a few examples of the ground truth data for the CityScapes dataset [Cor+16]. In our experiments, only 6.8% of the pixels are labeled as object boundaries in the pruned ground truth data.

3.3.4 Image Partition

Let V be the set of superpixels extracted from the output of the instance-aware edge detection block and $E \subseteq \binom{V}{2}$ be the set of neighboring superpixels, i.e., those having a common border.

With the methods described in Sections 3.3.2 and 3.3.3 we obtain:

- Log-probabilities $\alpha_{u,l}$ of all semantic labels $l \in \mathcal{L}$ (including background) for each superpixel $u \in V$.
- Log-probabilities $b_{u,v}$ for all pairs of neighbouring superpixels $\{u, v\} \in E$, for having a cutting edge.
- Prior log-probabilities of having a boundary between any two (also equal) semantic classes $\beta_{l,l'}$, for any two labels $l, l' \in \mathcal{L}$. In particular, the weight $\beta_{l,l}$ defines, how probable it is that two neighboring super-pixel have the same label l and belong to different instances. We set $\beta_{0,0}$ to $-\infty$, assuming there are no boundaries between superpixels labeled both as background.

We want to assign a single label to each superpixel and have close-contour boundaries, such that if two neighboring superpixels belong to different classes, there is always a boundary between them.

Our problem formulation consists of two components: (i) a conditional random field model [Kap+15] and (ii) a graph partition problem, known as MultiCut [CR93] or correlation clustering [BBC04]. In a certain sense, these two problems are coupled together in our formulation. Therefore, we first briefly describe each of them separately and afterwards consider their joint formulation.

Conditional Random Field (CRF). Let us, for now, assume that all $\beta_{l,l} = -\infty$, $l \in \mathcal{L}$, i.e., there can be no boundary between superpixels assigned the same label. In this case our problem is reduced to the following famous format: Let $G = (V, E)$ be an undirected graph. A finite set of labels $\mathcal{L}$ is associated with each node. With each label l in each node v a vector $\alpha_{v,l}$ is associated, which denotes *the score* of the label assigned to the node. Each pair of labels l, l' in neighbouring nodes $\{u, v\}$ is assigned a score

$$c_{u,v,l,l'} := \begin{cases} b_{u,v} + \beta_{l,l'}, & l \neq l' \\ 0, & l = l' \end{cases}$$

The vector $\mathbf{l} \in \mathcal{L}^{|V|}$ with coordinates l_u, $u \in V$ being labels assigned to each node is called *a labeling*. *The maximum a posteriori inference* problem for the CRF is defined above reads

$$\max_{\mathbf{l} \in L^{|V|}} \sum_{u \in V} \alpha_{u,l_u} + \sum_{uv \in E} c_{u,v,l_u,l_v} \,. \tag{3.2}$$

A solution to this problem is a usual (non-instance-aware) semantic segmentation, if we associate the graph nodes with superpixels and the graph edges will define their neighborhood.

For the MultiCut formulation below, we will require a different representation of the problem (3.2), in a form of *an integer quadratic problem*. Consider binary variables $x_{u,l} \in \{0, 1\}$ for each node $u \in V$ and label $l \in \mathcal{L}$. The equality $x_{u,l} = 1$ means that label l is assigned to the node u. The problem (3.2) now can be rewritten as follows:

$$\begin{aligned} &\max_{\mathbf{x}} \sum_{u \in V} \sum_{l \in \mathcal{L}} \alpha_{u,l} x_{u,l} + \sum_{uv \in E} \sum_{l \in \mathcal{L}} \sum_{l' \in \mathcal{L}} c_{u,v,l,l'} x_{u,l} x_{v,l'} \\ &\text{s.t.} \begin{cases} x_{u,l} \in \{0, 1\}, & u \in V, l \in \mathcal{L} \\ \sum_{l \in L} x_{u,l} = 1, & u \in V \,. \end{cases} \end{aligned} \tag{3.3}$$

The last constraint in (3.3) is added to guarantee that each node is assigned exactly one label. Although the problem (3.3) is NP-hard in general, it can be efficiently (and often exactly) solved for many practical instances appearing in computer vision, see [Kap+15] for an overview.

MultiCut Problem. Let us now assume a different situation, where all nodes have already got an assigned semantic label and all that we want is to *partition* each connected component (labeled with a single class) into connected regions corresponding to instances. Let us assume, for instance, that all superpixels of the component have a label l. This task has an elegant formulation as *a MultiCut* problem [CR93]:

Let $G = (V, E)$ be an undirected graph, with the scores $\theta_{u,v} := b_{u,v} + \beta_{l,l}$ assigned to the graph edges. Let also $\dot\cup$ stand for a disjoint union of sets. *The MultiCut* problem

(also known as correlation clustering) is to find a partitioning $(\Pi_1, \ldots, \Pi_k)$, $\Pi_i \subseteq V$, $V = \dot{\cup}_{i=1}^{k} \Pi_i$ of the graph vertices, such that the total score of edges connecting different components is maximized. The number k of components is not fixed but is determined by the algorithm itself. Although the problem is NP-hard in general, there are efficient approximate solvers for it, see e.g. [Bei+14; KL70; Keu+15].

In the following, we will require a different representation of the MultiCut problem, in form of *an integer linear problem*. To this end, we introduce a binary variable $y_e = y_{u,v} \in \{0, 1\}$ for each edge $e = \{u, v\} \in E$. This variable takes the value 1, if u and v belong to different components, i.e. $u \in \Pi_i$, $v \in \Pi_j$ for some $i \neq j$. Edges $\{u, v\}$ with $y_{u,v} = 1$ are called *cut* edges. The vector $\mathbf{y} \in \{0, 1\}^{|E|}$ with coordinates y_e, $e \in E$ is called *a MultiCut*. Let C be the set of all cycles of the graph G. It is a known result from combinatorial optimization [CR93] that the MultiCut problem can be written in the following form:

$$\max_{\mathbf{y} \in \{0,1\}^{|E|}} \sum_{\{u,v\} \in E} \theta_{u,v} y_{u,v} \,, \quad \text{s.t. } \forall C \, \forall e' \in C \colon \sum_{e \in C \setminus \{e'\}} y_e \geq y_{e'} \,. \tag{3.4}$$

Here, the objective directly maximizes the total score of the edges and the inequality constraints basically force each cycle to have none or at least two cut edges. These *cycle constraints* ensure that the set of cut edges actually defines a partitioning. Obviously, the cut edges correspond to boundaries in our application.

Our InstanceCut Problem. Let us combine both subproblems: We want to *jointly* infer both the semantic labels *and* the partitioning of each semantic segment, with each partition component defining an object instance. To this end, consider our InstanceCut problem (3.5)-(3.8) below:

$$\begin{aligned} \max_{\substack{\mathbf{x} \in \{0,1\}^{|V||\mathcal{L}|} \\ \mathbf{y} \in \{0,1\}^{|E|}}} & \sum_{u \in V} \sum_{l \in \mathcal{L}} \alpha_{u,l} x_{u,l} \\ & + w \sum_{uv \in E} \sum_{l \in \mathcal{L}} \sum_{l' \in \mathcal{L}} (b_{u,v} + \beta_{l,l'}) x_{u,l} x_{v,l'} y_{u,v} \end{aligned} \tag{3.5}$$

$$\sum_{l \in \mathcal{L}} x_{u,l} = 1, \; u \in V \tag{3.6}$$

$$\forall e' \in C \colon \sum_{e \in C \setminus \{e'\}} y_e \geq y_{e'} \tag{3.7}$$

$$\left.\begin{aligned} x_{u,l} - x_{v,l} \leq y_{uv} \\ x_{v,l} - x_{u,l} \leq y_{uv} \end{aligned}\right\}, \; \{u, v\} \in E, \; l \in \mathcal{L} \,. \tag{3.8}$$

Objective (3.5) and inequalities (3.6)-(3.7) are obtained directly from merging problems (3.3) and (3.4). We also introduced the parameter w that balances the modalities. Additional constraints (3.8) are required to guarantee that as soon as two neighboring nodes u and v are assigned different labels, the corresponding edge $y_{u,v}$ is cut and defines a part of an instance boundary. Two nodes u and u are assigned different labels if at most one of the variables $x_{u,l}$, $x_{v,l}$ takes value 1. In this case, the largest left-hand side of one of the inequalities (3.8) is equal to 1 and therefore $y_{u,v}$ must be cut. The problem related to (3.5)-(3.8) was considered in [Ham14] for foreground/background

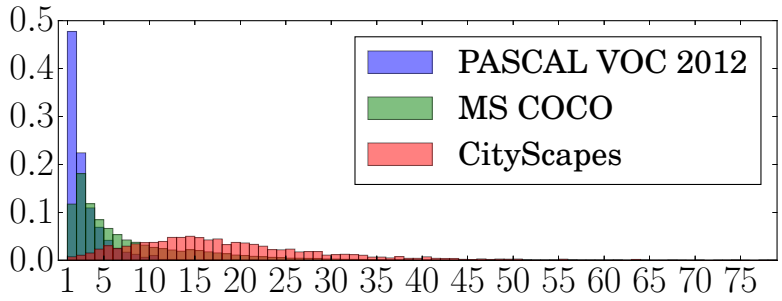


Figure 3.5: The histograms shows distribution of number of instances per image for different datasets. For illustrative reasons we cut long tails of CityScapes and MS COCO. We use CityScapes dataset since it contains significantly more instances per image.

segmentation.

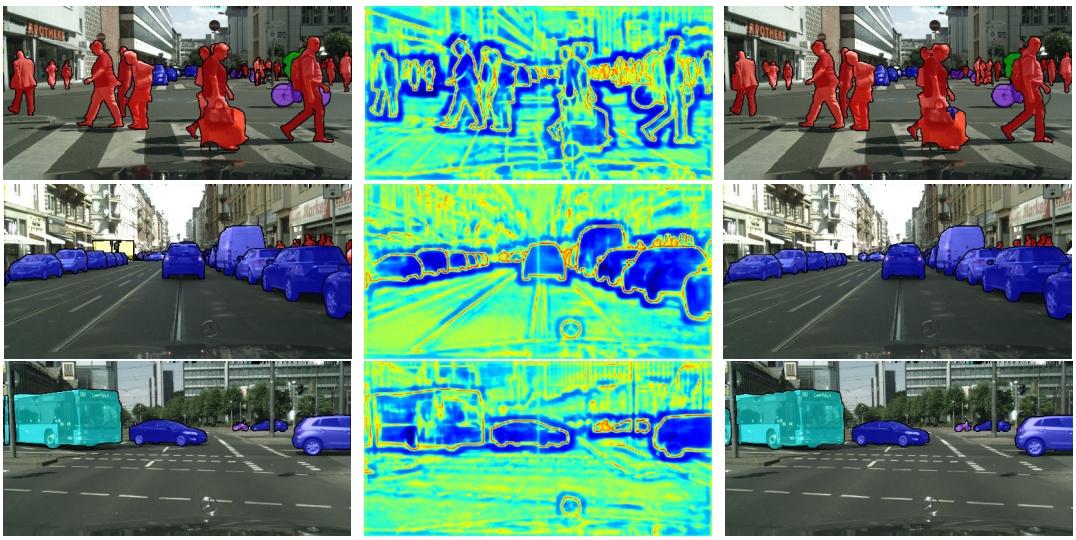
Although the problem (3.5)-(3.8) is NP-hard and it contains a lot of hard constraints, there exists an efficient approximate solver for it [Lev+17], which we used in our experiments. For solving the problem over 3000 nodes (superpixels) and 9 labels (segment classes) it required less than a second on average.

3.4 Experiments

Dataset. There are three main datasets with full annotation for the instance-aware semantic segmentation problem: PASCAL VOC2012 [Eve+15], MS COCO [Lin+14] and CityScapes [Cor+16]. We select the last one for our experimental evaluation for several reasons: (i) CityScapes has very fine annotation with precise boundaries for the annotated objects, whereas MS COCO has only coarse annotations, for some objects, that do not coincide with the true boundaries. Since our method uses an edge detector, it is important to to have precise object boundaries for training. (ii) The median number of instances per image in CityScapes is 16, whereas PASCAL VOC has 2 and MS COCO has 4. For this work a larger number is more interesting. The distribution of the number of instances per image for different datasets is shown in Fig. 3.5. (iii) Unlike other datasets, CityScapes' annotation is dense, i.e. all foreground objects are labeled.

The CityScape dataset has 5000 street-scene images recorded by car-mounted cameras: 2975 images for training, 500 for validation and 1525 for testing. There are 8 classes of objects that have an instance-level annotation in the dataset: person, rider, car, truck, bus, train, motorcycle, bicycle. All images have the size of 1024×2048 pixels.

Training details. For the semantic segmentation block in our framework we test two different networks, which have publicly available trained models for CityScapes: Dilation10 [YK16] and LRR-4x [GF16]. The latter is trained using the additional coarsely annotated data, available in CityScapes. Importantly, CityScapes has 19 different semantic segmentation classes (and only 8 out of them are considered for instance segmentation) and both networks were trained to segment all these classes. We do not retrain the networks and directly use the log-probabilities for the 8 semantic classes, which we require. For the background label we take the maximum over the



(a) Ground truth (b) Edges map (c) InstanceCut prediction

Figure 3.6: Qualitative results of InstanceCut framework. Left column contains input images with the highlighted ground truth instances. Middle column depicts per-pixel instance-aware edge log-probabilities and the last column shows the results of our approach. Note that in the last example the bus and a car in the middle are separated by a lamp-post, therefore, our method returns two instances for the objects.

log-probabilities of the remaining semantic classes.

As an initial semantic segmentation network for the instance-aware edge detection block we use Dilation10 [YK16] pre-trained on the CityScapes. We exactly follow the training procedure described in the original paper [YK16]. That is, we pre-train first the front-end module with the 2-layer perceptron on top. Then we pre-train the context module of the network separately and, finally, train the whole system end-to end. All the stages are trained with the same parameters as in [YK16]. In our experiments the 2-layer perceptron has 16 hidden neurons. On the validation set the trained detector achieves 97.2% AUC ROC.

Parameters w (see (3.5)) and $\beta_{l,l'}$, for all $l, l' \in \mathcal{L}$, in our InstanceCut formulation (3.5) are selected via 2-fold cross-validation. Instead of considering different $\beta_{l,l'}$ for all pairs of labels, we group them into two classes: 'big' and 'small'. All $\beta_{l,l'}$, where either l or l' corresponds to a (physically) big object, i.e., train, bus, or truck, are set to β_{big}. All other $\beta_{l,l'}$ are set to β_{small}. Therefore, our parameter space is only 3 dimensional and is determined by the parameters w, β_{small} and β_{big}.

Instance-level results - quantitative and qualitative. We evaluated our method using 4 metrics that are suggested by the CityScapes benchmark: AP, AP50%, AP100m and AP50m. We refer to the webpage of the benchmark for a detailed description.

The InstanceCut framework with Dilation10 [YK16] as the semantic segmentation block gives AP $= 14.8$ and AP$50\% = 30.7$ on the validation part of the dataset. When we replace Dilation10 by LRR-4x [GF16] for this block the performance improves to AP $= 15.8$ and AP$50\% = 32.4$, on the validation set.

Quantitative results for the test set are provided in Table 3.1. We compare our

Method	Metric	Mean	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle
MCG+R-CNN [Cor+16]	AP	4.6	1.3	0.6	10.5	6.1	9.7	5.9	1.7	0.5
Uhrig et al. [Uhr+16]	AP	8.9	**12.5**	**11.7**	22.5	3.3	5.9	3.2	6.9	**5.1**
InstanceCut	AP	**13.0**	10.0	8.0	**23.7**	**14.0**	**19.5**	**15.2**	**9.3**	4.7
MCG+R-CNN [Cor+16]	AP50%	12.9	5.6	3.9	26.0	13.8	26.3	15.8	8.6	3.1
Uhrig et al. [Uhr+16]	AP50%	21.1	**31.8**	**33.8**	37.8	7.6	12.0	8.5	20.5	**17.2**
InstanceCut	AP50%	**27.9**	28.0	26.8	**44.8**	**22.2**	**30.4**	**30.1**	**25.1**	15.7
MCG+R-CNN [Cor+16]	AP100m	7.7	2.6	1.1	17.5	10.6	17.4	9.2	2.6	0.9
Uhrig et al. [Uhr+16]	AP100m	15.3	**24.4**	**20.3**	36.4	5.5	10.6	5.2	10.5	**9.2**
InstanceCut	AP100m	**22.1**	19.7	14.0	**38.9**	**24.8**	**34.4**	**23.1**	**13.7**	8.0
MCG+R-CNN [Cor+16]	AP50m	10.3	2.7	1.1	21.2	14.0	25.2	14.2	2.7	1.0
Uhrig et al. [Uhr+16]	AP50m	16.7	**25.0**	**21.0**	40.7	6.7	13.5	6.4	11.2	**9.3**
InstanceCut	AP50m	**26.1**	20.1	14.6	**42.5**	**32.3**	**44.7**	**31.7**	**14.3**	8.2

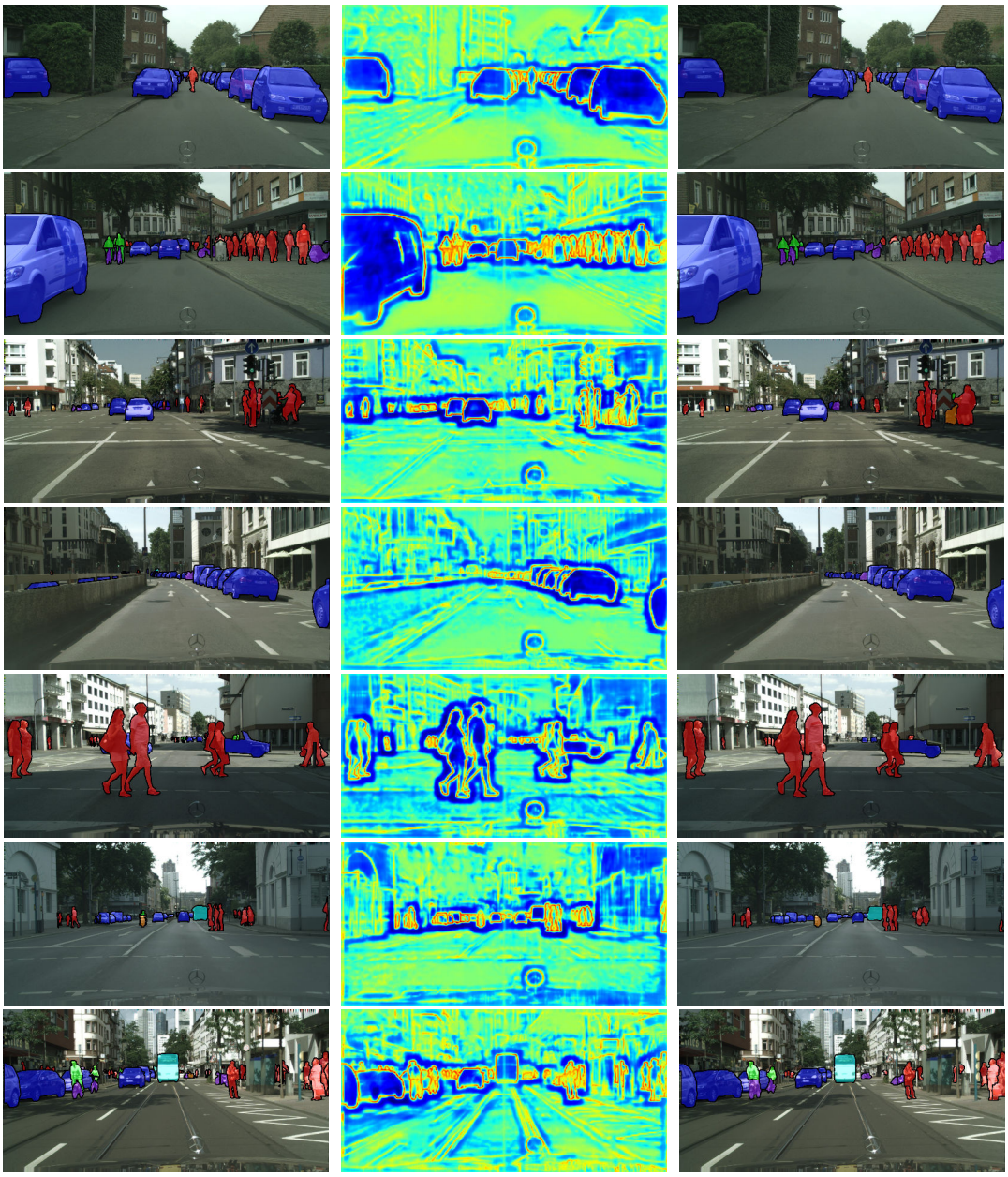
Table 3.1: CityScapes results. Instance-aware semantic segmentation results on the test set of CityScapes, given for each semantic class.

approach to previously published methods that have results for this dataset. Among them our method shows the best performance, despite its simplicity. A few new methods [He+17; Liu+17b; Liu+18] that outperfrom InstanceCut were proposed after its publication [Kir+17]. Note, however, that this methods use a much stronger backbone CNN architecture.

Fig. 3.7 contains the subset of difficult scenes where InstanceCut is able to predict most instances correctly. Fig. 3.8 contains failure cases of InstanceCut. The main sources of failure are: small objects that are far away from the camera, groups of people that are very close to camera and have heavy mutual occlusions, and occluded instances that have several disconnected visible parts.

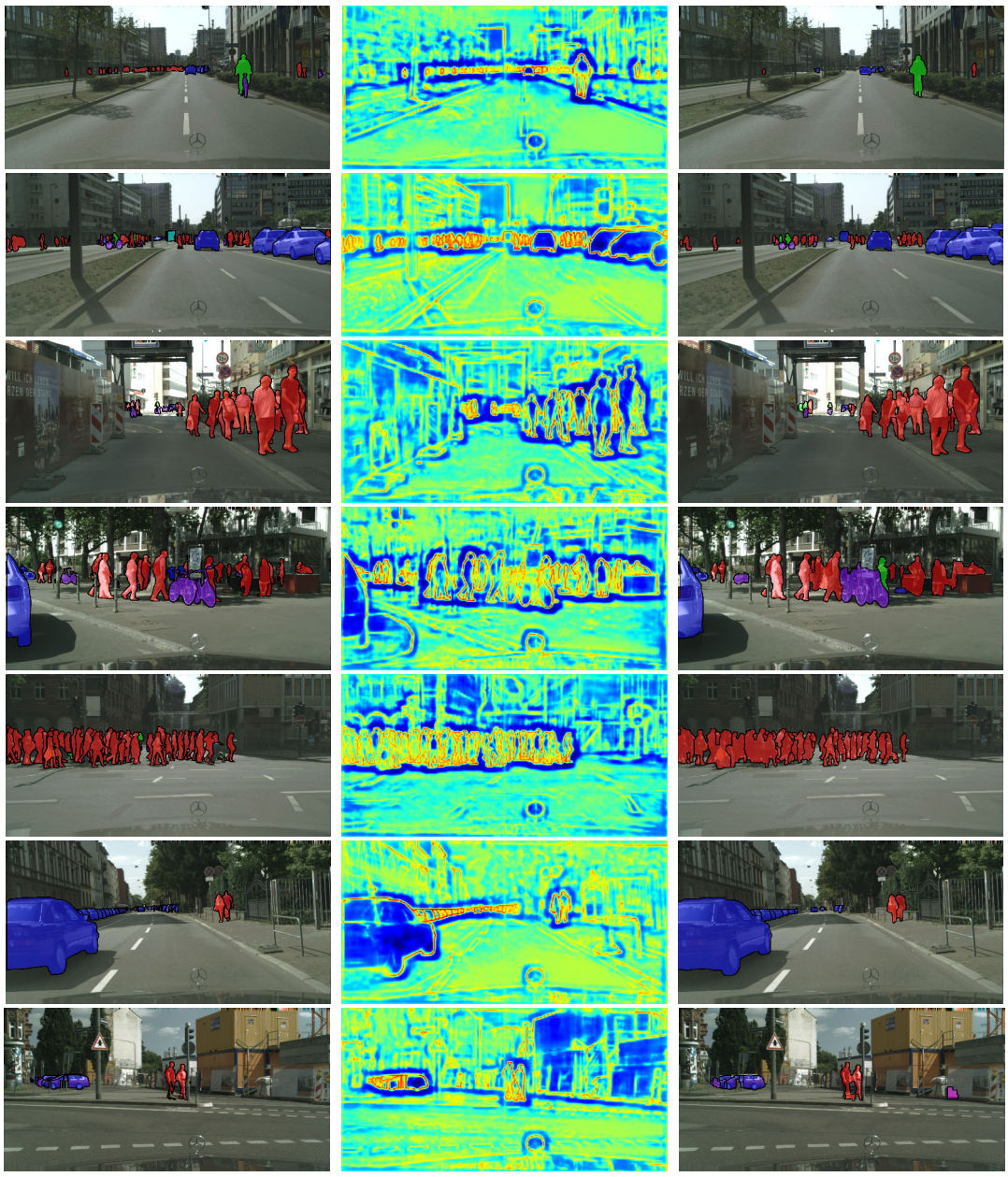
3.5 Discussion

We have proposed an alternative paradigm for instance-aware semantic segmentation. The paradigm represents the instance segmentation problem by a combination of two modalities: instance-agnostic semantic segmentation and instance-aware boundaries. We have presented a new framework that utilizes this paradigm. The modalities are produced by FCN networks. The standard FCN model is used for semantic segmentation, whereas a new architecture is proposed for object boundaries. The modalities are combined are combined by a novel MultiCut framework, which reasons globally about instances. The proposed framework achieves very promising results.



(a) Ground Truth (b) Edges Map (c) InstanceCut Prediction

Figure 3.7: Curated difficult scene, where InstanceCut performs well. The left column contains input images with ground truth instances highlighted. The middle column depicts per-pixel instance-aware edge log-probabilities and the last column shows the results of our approach.



(a) Ground Truth (b) Edges Map (c) InstanceCut Prediction

Figure 3.8: Failure cases. The left column contains input images with ground truth instances highlighted. The middle column depicts per-pixel instance-aware edge log-probabilities and the last column shows the results of our approach.

Chapter 4

Panoptic Segmentation

4.1 Introduction

In the early days of computer vision, *things* – countable objects such as people, animals, tools – received the dominant share of attention. Questioning the wisdom of this trend, Adelson [Ade01] elevated the importance of studying systems that recognize *stuff* – amorphous regions of similar texture or material such as grass, sky, road. This dichotomy between stuff and things persists to this day, reflected in both the division of visual recognition tasks and in the specialized algorithms developed for stuff and thing tasks.

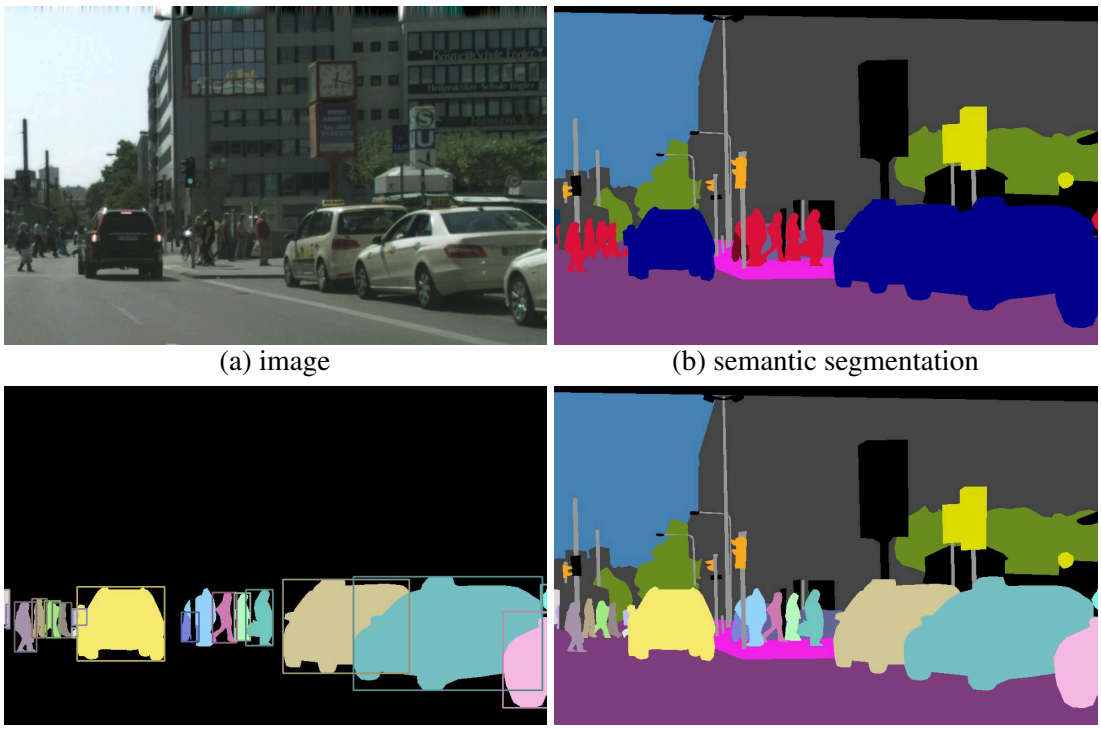
Studying stuff is most commonly formulated as a task known as *semantic segmentation*, see Figure 4.1b. As stuff is amorphous and uncountable, this task is defined as simply assigning a class label to each pixel in an image (note that semantic segmentation treats thing classes as stuff). In contrast, studying things is typically formulated as the task of *object detection* or *instance segmentation*, where the goal is to detect each object and delineate it with a bounding box or segmentation mask, respectively, see Figure 4.1c. While seemingly related, the datasets, details, and metrics for these two visual recognition tasks vary substantially.

The schism between semantic and instance segmentation has led to a parallel rift in the methods for these tasks. Stuff classifiers are usually built on fully convolutional nets [LSD15] with dilations [YK16; Che+17a] while object detectors often use object proposals [Hos+15] and are region-based [Ren+15; He+17]. Overall algorithmic progress on these tasks has been incredible in the past decade, yet, something important may be overlooked by focussing on these tasks in isolation.

A natural question emerges: *Can there be a reconciliation between stuff and things?* And what is the most effective design of a unified vision system that generates rich and coherent scene segmentations? These questions are particularly important given their relevance in real-world applications, such as autonomous driving or augmented reality.

Interestingly, while semantic and instance segmentation dominate current work, in the pre-deep learning era there was interest in the joint task described using various names such as *scene parsing* [TNL14], *image parsing* [Tu+05], or *holistic scene understanding* [YFU12]. Despite its practical relevance, this general direction is not currently popular, perhaps due to lack of appropriate metrics or recognition challenges.

In our work we aim to revive this direction. We propose a task that: *(1) encompasses*



(a) image (b) semantic segmentation

(c) instance segmentation (d) panoptic segmentation

Figure 4.1: For a given (a) image, we show *ground truth* for: (b) semantic segmentation (per-pixel class labels), (c) instance segmentation (per-object mask and class label), and (d) the proposed *panoptic segmentation* task (per-pixel class+instance labels). The PS task: (1) encompasses both stuff and thing classes, (2) uses a simple but general format, and (3) introduces a uniform evaluation metric for all classes. Panoptic segmentation generalizes both semantic and instance segmentation and we expect the unified task will present novel challenges and enable innovative new methods.

both stuff and thing classes, (2) uses a simple but general output format, and (3) introduces a uniform evaluation metric. To clearly disambiguate with previous work, we refer to the resulting task as *panoptic segmentation* (PS). The definition of 'panoptic' is "including everything visible in one view", in our context panoptic refers to a unified, global view of segmentation.

The **task format** we adopt for panoptic segmentation is simple: each pixel of an image must be assigned a semantic label and an instance id. Pixels with the same label and id belong to the same object; for stuff labels the instance id is ignored. See Figure 4.1d for a visualization. This format has been adopted previously, especially by methods that produce non-overlapping instance segmentations [Kir+17; Liu+17b; AT17]. We adopt it for our joint task that includes stuff and things.

A fundamental aspect of panoptic segmentation is the **task metric** used for evaluation. While numerous existing metrics are popular for either semantic or instance segmentation, these metrics are best suited either for stuff or things, respectively, but not both. We believe that the use of disjoint metrics is one of the primary reasons the community generally studies stuff and thing segmentation in isolation. To address this, we introduce the *panoptic quality* (PQ) metric in §4.4. PQ is *simple* and *informative* and most importantly can be used to measure the performance for both stuff and things

in a *uniform* manner. Our hope is that the proposed joint metric will aid in the broader adoption of the joint task.

The panoptic segmentation task encompasses both semantic and instance segmentation but introduces new algorithmic challenges. Unlike semantic segmentation, it requires differentiating individual object instances; this poses a challenge for fully convolutional nets. Unlike instance segmentation, object segments must be *non-overlapping*; this presents a challenge for region-based methods that operate on each object independently. Generating coherent image segmentations that resolve inconsistencies between stuff and things is an important step toward real-world uses.

As both the ground truth and algorithm format for PS must take on the same form, we can perform a detailed study of *human performance* on panoptic segmentation. This allows us to understand the PQ metric in more detail, including detailed breakdowns of recognition *vs*. segmentation and stuff *vs*. things performance. Moreover, measuring human PQ helps ground our understanding of machine performance. This is important as it will allow us to monitor performance saturations on various datasets for PS.

Finally we perform an initial study of machine performance for PS. To do so, we define a simple and likely suboptimal heuristic that combines the output of two *independent* systems for semantic and instance segmentation via a series of post-processing steps that merges their outputs (in essence, a sophisticated form of non-maximum suppression). Our heuristic establishes a baseline for PS and gives us insights into the main algorithmic challenges it presents.

We study both human and machine performance on three popular segmentation datasets that have both stuff and things annotations. This includes the Cityscapes [Cor+16], ADE20k [Zho+17], and Mapillary Vistas [Neu+17] datasets. For each of these datasets, we obtained results of state-of-the-art methods directly from the challenge organizers. In the future we will extend our analysis to COCO [Lin+14] on which stuff is being annotated [CUF18]. Together our results on these datasets form a solid foundation for the study of both human and machine performance on panoptic segmentation.

We are currently working with challenge organizers from the COCO [Lin+14], Vistas [Neu+17], and ADE20k [Zho+17] datasets to feature a panoptic segmentation track. We believe including a PS track alongside existing instance and semantic segmentation tracks on these popular recognition datasets will help lead to a broader adoption of the proposed joint task.

4.2 Related Work

Novel datasets and tasks have played a key role throughout the history of computer vision. They help catalyze progress and enable breakthroughs in our field, and just as importantly, they help us measure and recognize the progress our community is making. For example, ImageNet [Rus+15] helped drive the recent popularization of deep learning techniques for visual recognition [KSH12] and exemplifies the potential transformational power that datasets and tasks can have. Our goals for introducing the panoptic segmentation task are similar: to challenge our community, to drive research in novel directions, and to enable both expected and unexpected innovation. We review related tasks next.

Object detection tasks. Early work on face detection using ad-hoc datasets (*e.g.*, [VML94; VJ01]) helped popularize bounding-box object detection. Later, pedestrian detection datasets [Dol+12] helped drive progress in the field. The PASCAL VOC dataset [Eve+15] upgraded the task to a more diverse set of general object classes on more challenging images. More recently, the COCO dataset [Lin+14] pushed detection towards the task of instance segmentation. By framing this task and providing a high-quality dataset, COCO helped define a new and exciting research direction and led to many recent breakthroughs in instance segmentation [PCD15; Li+17; He+17]. Our general goals for panoptic segmentation are similar.

Semantic segmentation tasks. Semantic segmentation datasets have a rich history [Sho+06; LYT11; Eve+15] and helped drive key innovations (*e.g.*, fully convolutional nets [LSD15] were developed using [LYT11; Eve+15]). These datasets contain both stuff and thing classes, but don't distinguish individual object instances. Recently the field has seen numerous new segmentation datasets including Cityscapes [Cor+16], ADE20k [Zho+17], and Mapillary Vistas [Neu+17]. These datasets actually support both semantic and instance segmentation, and each has opted to have a separate track for the two tasks. Importantly, they contain all of the information necessary for PS. In other words, *the panoptic segmentation task can be bootstrapped on these datasets without any new data collection.*

Multitask learning. With the success of deep learning for many visual recognition tasks, there has been substantial interest in *multitask learning* approaches that have broad competence and can solve multiple diverse vision problems in a single framework [Kok17; Mal+16; Mis+16]. *E.g.*, UberNet [Kok17] solves multiple low to high-level visual tasks, including object detection and semantic segmentation, using a single network. While there is significant interest in this area, we emphasize that panoptic segmentation is *not* a multitask problem but rather a single, *unified* view of image segmentation. Specifically, the multitask setting allows for independent and potentially inconsistent outputs for stuff and things, while PS requires a single coherent scene segmentation.

Joint segmentation tasks. In the pre-deep learning era, there was substantial interest in generating coherent scene interpretations. The seminal work on image parsing [Tu+05] proposed a general bayesian framework to jointly model segmentation, detection, and recognition. Later, approaches based on graphical models studied consistent stuff and thing segmentation [YFU12; TL13; TNL14; Sun+14]. While these methods shared a common motivation, there was no agreed upon task definition, and different output formats and varying evaluation metrics were used, including separate metrics for evaluating results on stuff and thing classes. In recent years this direction has become less popular, perhaps for these reasons.

In our work we aim to revive this general direction, but in contrast to earlier work, we focus on the task itself. Specifically, as discussed, PS: (1) addresses both stuff and thing classes, (2) uses a simple format, and (3) introduces a uniform metric for both stuff and things. Previous work on joint segmentation uses varying formats and disjoint metrics for evaluating stuff and things. Methods that generate non-overlapping

instance segmentations [Kir+17; BU17; Liu+17b; AT17] use the same format as PS, but these methods typically only address thing classes. By addressing both stuff and things, using a simple format, and introducing a uniform metric, we hope to encourage broader adoption of the joint task.

Amodal segmentation task. In [Zhu+17] objects are annotated *amodally*: the full extent of each region is marked, not just the visible. Our work focuses on segmentation of all *visible* regions, but an extension of panoptic segmentation to the amodal setting is an interesting direction for future work.

4.3 Panoptic Segmentation Format

Task format. The format for panoptic segmentation is simple to define. Given a predetermined set of L semantic classes encoded by $\mathcal{L} := \{1, \ldots, L\}$, the task requires a *panoptic segmentation algorithm* to map each pixel i of an image to a pair $(l_i, z_i) \in \mathcal{L} \times \mathbb{N}$, where l_i represents the semantic class of pixel i and z_i represents its instance id. Instances, not pixels, are the atomic units of output produced by the algorithm that will be used in a matching process for evaluation (described later). Ground truth annotations for an image are encoded in an identical manner.

Stuff and thing labels. The semantic label set consists of subsets $\mathcal{L}^{\text{St}}$ and $\mathcal{L}^{\text{Th}}$, such that $\mathcal{L} = \mathcal{L}^{\text{St}} \cup \mathcal{L}^{\text{Th}}$ and $\mathcal{L}^{\text{St}} \cap \mathcal{L}^{\text{Th}} = \emptyset$. These subsets correspond to *stuff* and *thing* labels, respectively. When a pixel is labeled with $l_i \in \mathcal{L}^{\text{St}}$, its corresponding instance id z_i is irrelevant. That is, for stuff classes all pixels belong to the same instance (*e.g.*, the same *sky*). Otherwise, all pixels with the same (l_i, z_i) assignment, where $l_i \in \mathcal{L}^{\text{Th}}$, belong to the same instance (*e.g.*, the same *car*), and conversely, all pixels belonging to a single instance must have the same (l_i, z_i). The selection of which classes are stuff *vs.* things is a design choice left to the creator of the dataset, just as in previous datasets.

Relationship to semantic segmentation. The PS task format is a strict generalization of the format for semantic segmentation. Indeed, both tasks require each pixel in an image to be assigned a semantic label. If the ground truth does not specify instances, or all classes are stuff, then the task formats are identical (although the task metrics differ). In addition, inclusion of thing classes, which may have multiple instances per image, differentiates the tasks.

Relationship to instance segmentation. The instance segmentation task requires a method to segment each object instance in an image. However, it allows overlapping segments, whereas the panoptic segmentation task permits only one semantic label and one instance id to be assigned to each pixel. Hence, for PS, no overlaps are possible by construction. In the next section we show that this difference plays an important role in performance evaluation.

Confidence scores. Like semantic segmentation, but unlike instance segmentation, we do *not* require confidence scores associated with each segment for PS. This makes

the panoptic task *symmetric* with respect to humans and machines: both must generate the same type of image annotation. It also makes evaluating human performance for PS simple. This is in contrast to instance segmentation, which is not easily amenable to such a study as human annotators do not provide explicit confidence scores (though a single precision/recall point may be measured). We note that confidence scores give downstream systems more information, which can be useful, so it may still be desirable to have a PS algorithm generate confidence scores in certain settings.

4.4 Panoptic Segmentation Metric

In this section we introduce a new metric for panoptic segmentation. We begin by noting that existing metrics are specialized for either semantic or instance segmentation and cannot be used to evaluate the joint task involving both stuff and thing classes. Previous work on joint segmentation sidestepped this issue by evaluating stuff and thing performance using independent metrics (*e.g*. [YFU12; TL13; TNL14; Sun+14]). However, this introduces challenges in algorithm development, makes comparisons more difficult, and hinders communication. We hope that introducing a unified metric for stuff and things will encourage the study of the unified task.

Before going into further details, we start by identifying the following desiderata for a suitable metric for PS:

Completeness. The metric should treat stuff and thing classes in a uniform way, capturing all aspects of the task.

Interpretability. We seek a metric with identifiable meaning that facilitates communication and understanding.

Simplicity. In addition, the metric should be simple to define and implement. This improves transparency and allows for easy reimplementation. Related to this, the metric should be efficient to compute to enable rapid evaluation.

Guided by these principles, we propose a new *panoptic quality* (PQ) metric. PQ measures the quality of a predicted panoptic segmentation relative to the ground truth. It involves two steps: (1) segment matching and (2) PQ computation given the matches. We describe each step next then return to a comparison to existing metrics.

4.4.1 Segment Matching

We specify that a predicted segment and a ground truth segment can match only if their intersection over union (IoU) is strictly greater than 0.5. This requirement, together with the non-overlapping property of a panoptic segmentation, gives a *unique matching*: there can be at most one predicted segment matched with each ground truth segment.

Theorem 5. *Given a predicted and ground truth panoptic segmentation of an image, each ground truth segment can have at most one corresponding predicted segment with IoU strictly greater than 0.5 and vice verse.*

Proof. Let g be a ground truth segment and p_1 and p_2 be two predicted segments. By

definition, $p_1 \cap p_2 = \emptyset$ (they do not overlap). Since $|p_i \cup g| \geq |g|$, we get the following:

$$\text{IoU}(p_i, g) = \frac{|p_i \cap g|}{|p_i \cup g|} \leq \frac{|p_i \cap g|}{|g|} \quad \text{for } i \in \{1, 2\}\,.$$

Summing over i, and since $|p_1 \cap g| + |p_2 \cap g| \leq |g|$ due to the fact that $p_1 \cap p_2 = \emptyset$, we get:

$$\text{IoU}(p_1, g) + \text{IoU}(p_2, g) \leq \frac{|p_1 \cap g| + |p_2 \cap g|}{|g|} \leq 1\,.$$

Therefore, if $\text{IoU}(p_1, g) > 0.5$, then $\text{IoU}(p_2, g)$ has to be smaller than 0.5. Reversing the role of p and g can be used to prove that only one ground truth segment can have IoU with a predicted segment strictly greater than 0.5. □

The requirement that matches must have IoU greater than 0.5, which in turn yields the unique matching theorem, achieves two of our desired properties. First, it is *simple* and efficient as correspondences are unique and trivial to obtain. Second, it is *interpretable* and easy to understand (and does not require solving a complex matching problem as is commonly the case for these types of metrics [Har+14; Yan+12]).

Note that due to the uniqueness property, for IoU > 0.5, any reasonable matching strategy (including greedy and optimal) will yield an identical matching. For smaller IoU other matching techniques would be required; however, in the experiments we will show that lower thresholds are unnecessary as matches with IoU ≤ 0.5 are rare in practice.

4.4.2 Panoptic Quality (PQ) Computation

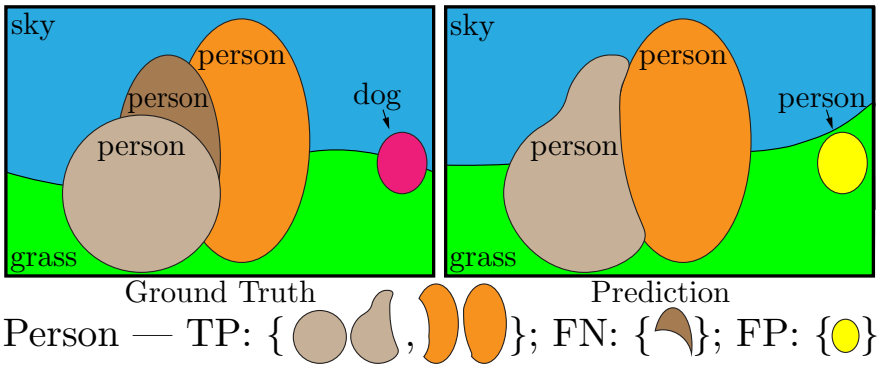


Figure 4.2: Toy illustration of ground truth and predicted panoptic segmentations of an image. Pairs of segments of the same color have IoU larger than 0.5 and are therefore matched. We show how the segments for the *person* class are partitioned into true positives TP, false negatives FN, and false positives FP.

We calculate PQ for each class independently and average over classes. This makes PQ insensitive to class imbalance. For each class, the unique matching splits the predicted and ground truth segments into three sets: true positives (TP), false positives (FP), and false negatives (FN), representing matched pairs of segments, unmatched

predicted segments, and unmatched ground truth segments, respectively. An example is illustrated in Figure 4.2. Given these three sets, PQ is defined as:

$$\text{PQ} = \frac{\sum_{(p,g)\in TP} \text{IoU}(p,g)}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|}. \tag{4.1}$$

PQ is intuitive after inspection: $\frac{1}{|TP|}\sum_{(p,g)\in TP} \text{IoU}(p,g)$ is simply the average IoU of matched segments, while $\frac{1}{2}|FP| + \frac{1}{2}|FN|$ is added to the denominator to penalize segments without matches. Note that all segments receive equal importance regardless of their area. Furthermore, if we multiply and divide PQ by the size of the TP set, then PQ can be seen as the multiplication of a *segmentation quality* (SQ) term and a *recognition quality* (RQ) term:

$$\text{PQ} = \underbrace{\frac{\sum_{(p,g)\in TP} \text{IoU}(p,g)}{|TP|}}_{\text{segmentation quality (SQ)}} \times \underbrace{\frac{|TP|}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|}}_{\text{recognition quality (RQ)}}. \tag{4.2}$$

Written this way, RQ is the familiar F_1 score [VR79] widely used for quality estimation in detection settings [MFM04]. SQ is simply the average IoU of matched segments. We find the decomposition of PQ = SQ $\times$ RQ to provide insight for analysis. We note, however, that the two values are not independent since SQ is measured only over matched segments.

Our definition of PQ achieves our desiderata. It measures performance of all classes in a uniform way using a simple and interpretable formula. We conclude by discussing how we handle void regions and groups of instances [Lin+14].

Void labels. There are two sources of void labels in the ground truth: (a) out of class pixels and (b) ambiguous or unknown pixels. As often we cannot differentiate these two cases, we don't evaluate predictions for void pixels. Specifically: (1) during matching, all pixels in a predicted segment that are labeled as void in the ground truth are removed from the prediction and do not affect IoU computation, and (2) after matching, unmatched predicted segments that contain a fraction of void pixels over the matching threshold are removed and do not count as false positives. Finally, outputs may also contain void pixels; these do not affect evaluation.

Group labels. A common annotation practice [Cor+16; Lin+14] is to use a group label instead of instance ids for adjacent instances of the same semantic class if accurate delineation of each instance is difficult. For computing PQ: (1) during matching, group regions are not used, and (2) after matching, unmatched predicted segments that contain a fraction of pixels from a group of the same class over the matching threshold are removed and do not count as false positives.

4.4.3 Comparison to Existing Metrics

We conclude by comparing PQ to existing metrics for semantic and instance segmentation.

Semantic segmentation metrics. Common metrics for semantic segmentation include pixel accuracy, mean accuracy, and IoU [LSD15]. These metrics are computed

based only on pixel outputs/labels and completely ignore object-level labels. For example, IoU is the ratio between correctly predicted pixels and total number of pixels in either the prediction or ground truth for each class. As these metrics ignore instance labels, they are not well suited for evaluating thing classes. Finally, please note that IoU for semantic segmentation is distinct from our segmentation quality (SQ), which is computed as the average IoU over *matched segments*.

Instance segmentation metrics. The standard metric for instance segmentation is Average Precision (AP) [Lin+14; Har+14]. AP requires each object segment to have a confidence score to estimate a precision/recall curve. Note that while confidence scores are quite natural for object detection, they are not used for semantic segmentation. Hence, AP cannot be used for measuring the output of semantic segmentation, or likewise of PS (see also the discussion of confidences in §4.3).

Panoptic quality. PQ treats all classes (stuff and things) in a uniform way. We note that while decomposing PQ into SQ and RQ is helpful with interpreting results, PQ is *not* a combination of semantic and instance segmentation metrics. Rather, SQ and RQ are computed for every class (stuff and things), and measure segmentation and recognition quality, respectively. PQ thus unifies evaluation over all classes. We support this claim with rigorous experimental evaluation of PQ in §4.7, including comparisons to IoU and AP for semantic and instance segmentation, respectively.

4.5 Panoptic Segmentation Datasets

To our knowledge only three public datasets have both dense semantic and instance segmentation annotations: Cityscapes [Cor+16], ADE20k [Zho+17], and Mapillary Vistas [Neu+17]. We use all three datasets for panoptic segmentation. In addition, in the future we will extend our analysis to COCO [Lin+14] on which stuff is currently being annotated [CUF18][1].

Cityscapes [Cor+16] has 5000 images (2975 train, 500 val, and 1525 test) of egocentric driving scenarios in urban settings. It has dense pixel annotations (97% coverage) of 19 classes among which 8 have instance-level segmentations.

ADE20k [Zho+17] has over 25k images (20k train, 2k val, 3k test) that are densely annotated with an open-dictionary label set. For the 2017 Places Challenge[2], 100 thing and 50 stuff classes that cover 89% of all pixels are selected. We use this closed vocabulary in our study.

Mapillary Vistas [Neu+17] has 25k street-view images (18k train, 2k val, 5k test) in a wide range of resolutions. The 'research edition' of the dataset is densely annotated (98% pixel coverage) with 28 stuff and 37 thing classes.

[1] In addition to stuff annotations being incomplete, COCO instance segmentations contain overlaps. We plan on collecting depth ordering for all pairs of overlapping instances in COCO to resolve these overlaps.

[2] `http://placeschallenge.csail.mit.edu`

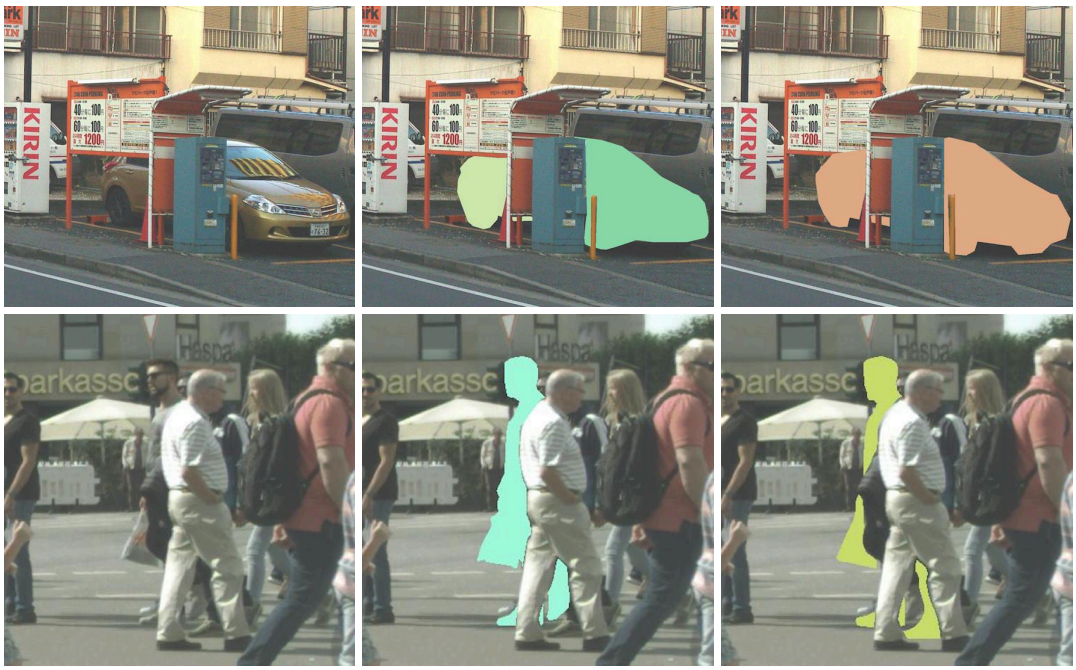


Figure 4.3: **Segmentation flaws.** Images are zoomed and cropped. Top row (Vistas image): both annotators identify the object as a car, however, one splits the car into two cars. Bottom row (Cityscapes image): the segmentation is genuinely ambiguous.

4.6 Human Performance Study

One advantage of panoptic segmentation is that it enables measuring human performance. Aside from this being interesting as an end in itself, human performance studies allow us to understand the task in detail, including details of our proposed metric and breakdowns of human performance along various axes. This gives us insight into intrinsic challenges posed by the task without biasing our analysis by algorithmic choices. Furthermore, human studies help ground machine performance (discussed in §4.7) and allow us to calibrate our understanding of the task.

Human annotations. To enable human performance analysis, dataset creators graciously supplied us with 30 doubly annotated images for Cityscapes, 64 for ADE20k, and 46 for Vistas. For Cityscapes and Vistas, the images are annotated independently by different annotators. ADE20k is annotated by a single well-trained annotator who labeled the same set of images with a gap of six months. To measure panoptic quality (PQ) for human annotators, we treat one annotation for each image as ground truth and the other as the prediction. Note that the PQ is symmetric w.r.t. the ground truth and prediction, so order is unimportant.

Human performance. First, Table 4.1 shows human performance on each dataset, along with the decomposition of PQ into segmentation quality (SQ) and recognition quality (RQ). As expected, humans are not perfect at this task, which is consistent with studies of annotation quality from [Cor+16; Zho+17; Neu+17]. Visualizations of human segmentation and classification errors are shown in Figures 4.3 and 4.4, respectively.

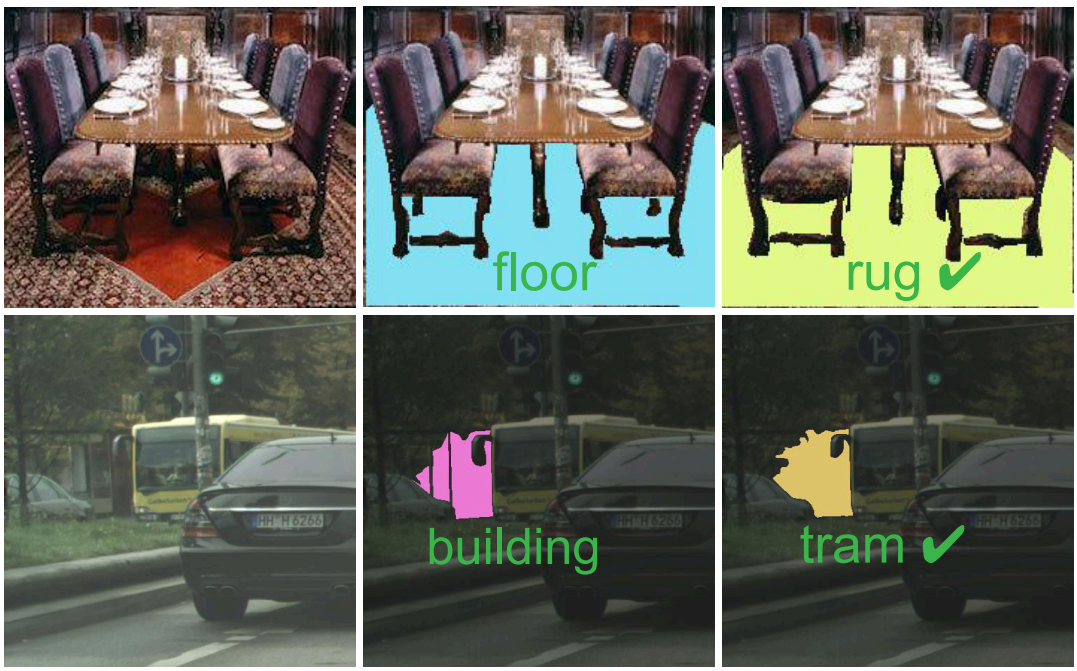


Figure 4.4: **Classification flaws.** Images are zoomed and cropped. Top row (ADE20k image): simple misclassification. Bottom row (Cityscapes image): the scene is extremely difficult, tram is the correct class for the segment. Many errors are difficult to resolve.

	PQ	PQ^{St}	PQ^{Th}	SQ	SQ^{St}	SQ^{Th}	RQ	RQ^{St}	RQ^{Th}
Cityscapes	69.7	71.3	67.4	84.2	84.4	83.9	82.1	83.4	80.2
ADE20k	67.1	70.3	65.9	85.8	85.5	85.9	78.0	82.4	76.4
Vistas	57.5	62.6	53.4	79.5	81.6	77.9	71.4	76.0	67.7

Table 4.1: **Human performance for stuff *vs*. things**. Panoptic, segmentation, and recognition quality (PQ, SQ, RQ) averaged over classes (PQ=SQ×RQ per class) are reported as percentages. Perhaps surprisingly, we find that human performance on each dataset is relatively similar for both stuff and things.

We note that Table 4.1 establishes a measure of annotator agreement on each dataset, *not* an upper bound on human performance. We further emphasize that numbers are not comparable across datasets and should not be used to assess dataset quality. The number of classes, percent of annotated pixels, and scene complexity vary across datasets, each of which significantly impacts annotation difficulty.

Stuff *vs*. things. PS requires segmentation of both stuff and things. In Table 4.1 we also show PQ^{St} and PQ^{Th} which is the PQ averaged over stuff classes and thing classes, respectively. For Cityscapes and ADE20k human performance for stuff and things are close, on Vistas the gap is a bit larger. Overall, this implies stuff and things have similar difficulty, although thing classes are somewhat harder. In Figure 4.5 we show PQ for every class in each dataset, sorted by PQ. Observe that stuff and things classes distribute fairly evenly. This implies that the proposed metric strikes a good balance

	PQ^S	PQ^M	PQ^L	SQ^S	SQ^M	SQ^L	RQ^S	RQ^M	RQ^L
Cityscapes	35.5	63.5	86.2	67.6	80.2	89.7	52.2	78.7	95.9
ADE20k	53.7	68.5	79.5	78.0	84.3	88.4	69.0	81.2	89.6
Vistas	37.1	47.9	69.9	70.2	76.6	83.0	53.7	62.7	83.4

Table 4.2: **Human performance *vs*. scale**, for small (S), medium (M) and large (L) objects. Scale plays a large role in determining human accuracy for panoptic segmentation. On large objects both SQ and RQ are above 80 on all datasets, while for small objects RQ drops precipitously. SQ for small objects is quite reasonable.

and, indeed, is successful at unifying the stuff and things segmentation tasks without either dominating the error.

Small *vs*. large objects. To analyze how PQ varies with object size we partition the datasets into small (S), medium (M), and large (L) objects by considering the smallest 25%, middle 50%, and largest 25% of objects in each dataset, respectively. In Table 4.2, we see that for large objects human performance for all datasets is quite good. For small objects, RQ drops significantly implying human annotators often have a hard time finding small objects. However, if a small object is found, it is segmented relatively well.

IoU threshold. By enforcing an overlap greater than 0.5 IoU, we are given a unique matching by Theorem 5. However, is the 0.5 threshold reasonable? An alternate strategy is to use no threshold and perform the matching by solving a maximum weighted bipartite matching problem [Wes01]. The optimization will return a matching that maximizes the sum of IoUs of the matched segments. We perform the matching using this optimization and plot the cumulative density functions of the match overlaps in Figure 4.6. Less than 16% of the matches have IoU overlap less than 0.5, indicating that relaxing the threshold should have minor effect.

To verify this intuition, in Figure 4.7 we show PQ computed for different IoU thresholds. Notably, the difference in PQ for IoU of 0.25 and 0.5 is relatively small, especially compared to the gap between IoU of 0.5 and 0.75, where the change in PQ is larger. Furthermore, many matches at lower IoU are false matches. Therefore, given that the matching for IoU of 0.5 is not only unique, but also simple and intuitive, we believe that the default choice of 0.5 is reasonable.

SQ *vs*. RQ balance. Our RQ definition is equivalent to the F_1 score. However, other choices are possible. Inspired by the generalized F_β score [VR79], we can introduce a parameter α that enables tuning the penalty for recognition errors:

$$\mathrm{RQ}^\alpha = \frac{|TP|}{|TP| + \alpha|FP| + \alpha|FN|}\,. \tag{4.3}$$

By default α is 0.5. Lowering α reduces the penalty of unmatched segments and thus increases RQ (SQ is not affected). Since PQ=SQ$\times$RQ, this changes the relative effect of PS *vs*. RQ on the final PQ metric. In Figure 4.8 we show SQ and RQ for various

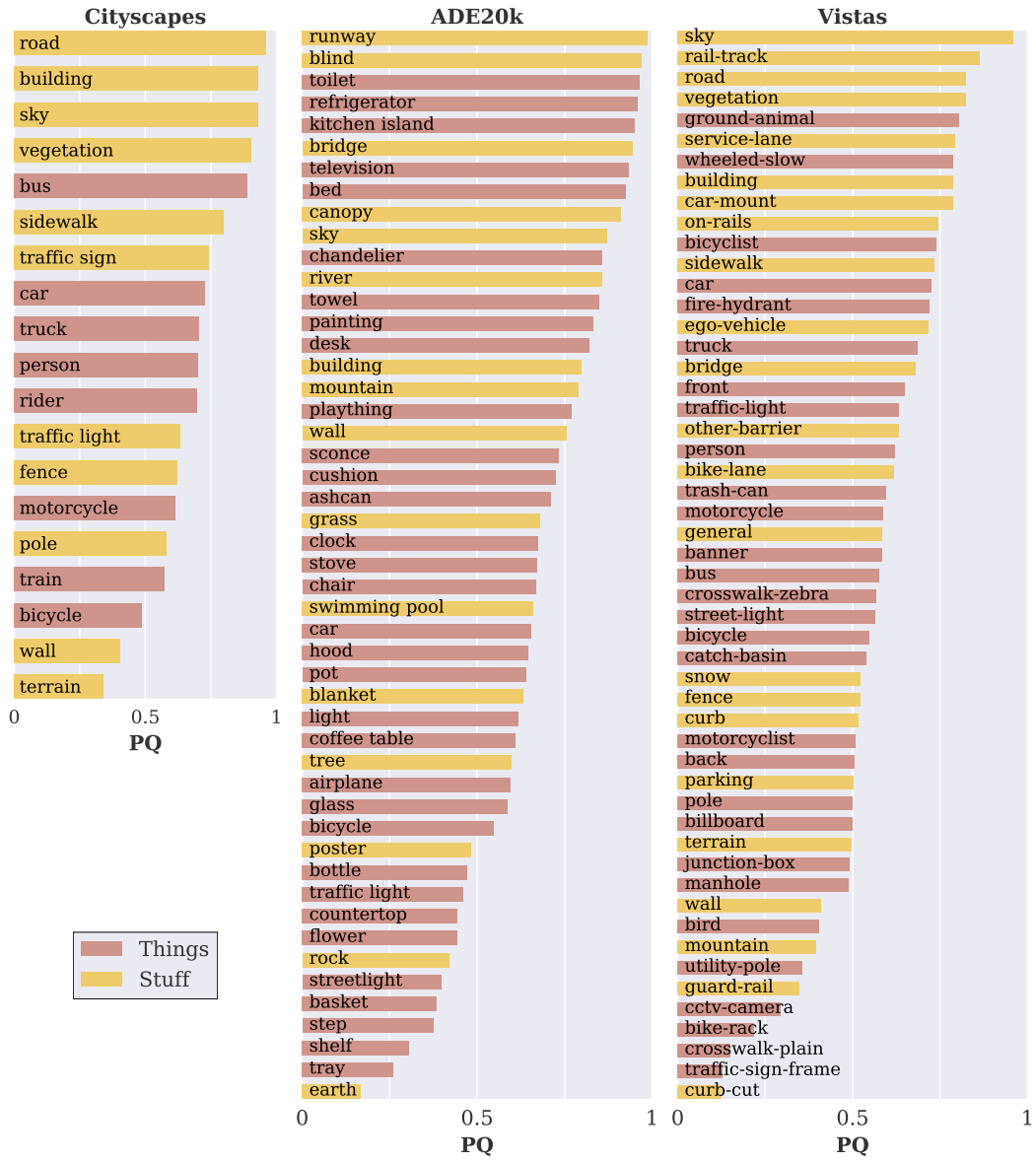


Figure 4.5: **Per-Class Human performance, sorted by PQ**. Thing classes are shown in red, stuff classes in orange (for ADE20k every other class is shown, classes without matches in the dual-annotated tests sets are omitted). Things and stuff are distributed fairly evenly, implying PQ balances their performance.

α. The default α strikes a good balance between SQ and RQ. In principle, altering α can be used to balance the influence of segmentation and recognition errors on the final metric. In a similar spirit, one could also add a parameter β to balance influence of FPs *vs*. FNs.

4.7 Machine Performance Baselines

We now present simple machine baselines for panoptic segmentation. We are interested in three questions: (1) How do heuristic combinations of top-performing instance and semantic segmentation systems perform on panoptic segmentation? (2) How does PQ

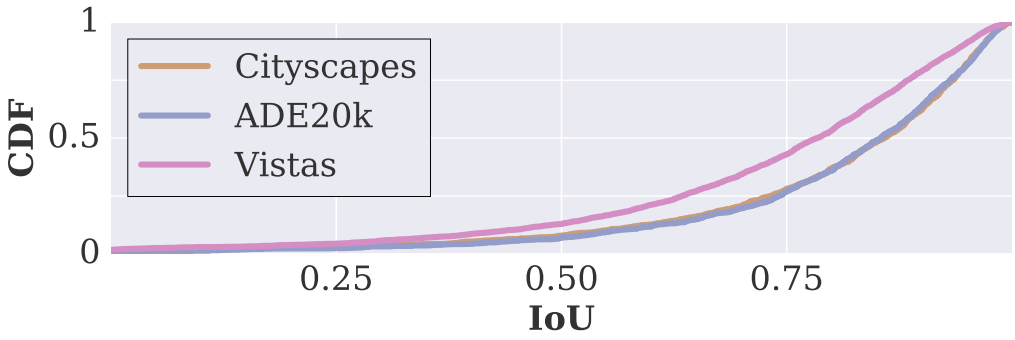


Figure 4.6: **Cumulative density functions of overlaps** for matched segments in three datasets when matches are computed by solving a maximum weighted bipartite matching problem [Wes01]. After matching, less than 16% of matched objects have IoU below 0.5.

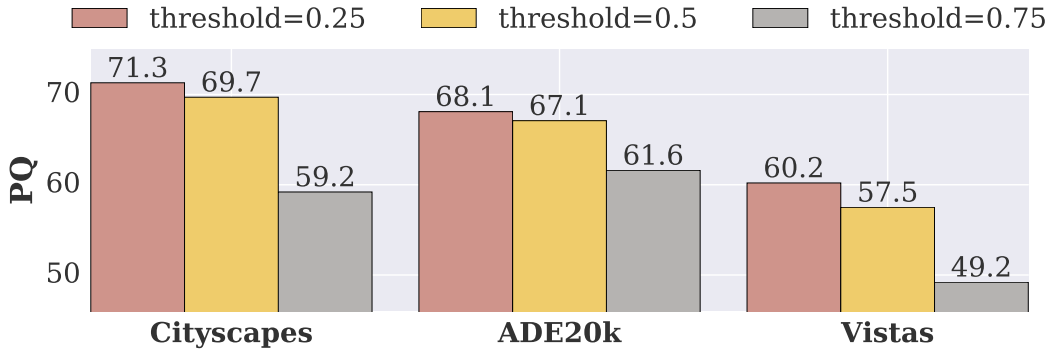


Figure 4.7: **Human performance for different IoU thresholds.** The difference in PQ using a matching threshold of 0.25 *vs*. 0.5 is relatively small. For IoU of 0.25 matching is obtained by solving a maximum weighted bipartite matching problem. For a threshold greater than 0.5 the matching is unique and much easier to obtain.

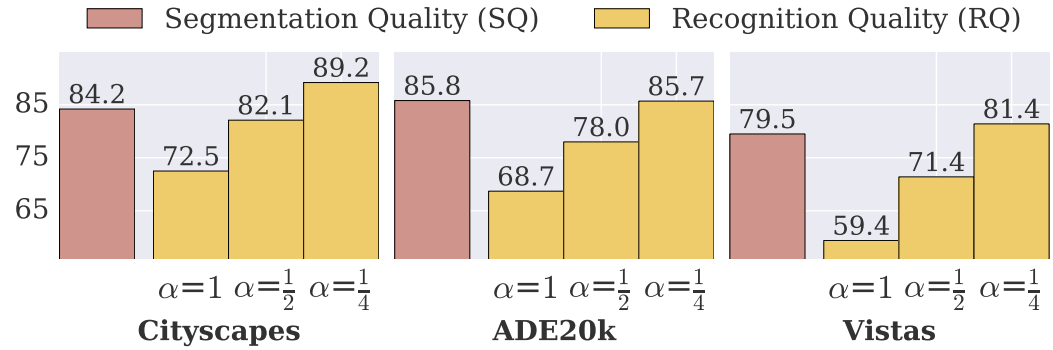


Figure 4.8: **SQ *vs*. RQ** for different α, see (4.3). Lowering α reduces the penalty of unmatched segments and thus increases the reported RQ (SQ is not affected). We use α of 0.5 throughout but by tuning α one can balance the influence of SQ and RQ in the final metric.

Cityscapes	AP	AP^{NO}	PQ^{Th}	SQ^{Th}	RQ^{Th}
Mask R-CNN+COCO [He+17]	**36.4**	**33.1**	**54.1**	**79.4**	**67.9**
Mask R-CNN [He+17]	31.5	28.0	49.6	78.7	63.0
ADE20k	AP	AP^{NO}	PQ^{Th}	SQ^{Th}	RQ^{Th}
Megvii [Luo+17]	**30.1**	**24.8**	**41.1**	**81.6**	**49.6**
G-RMI [FKM17]	24.6	20.6	35.3	79.3	43.2

Table 4.3: **Machine results on instance segmentation** (stuff classes ignored). Non-overlapping predictions are obtained using the proposed heuristic. AP^{NO} is AP of the non-overlapping predictions. As expected, removing overlaps harms AP as detectors benefit from predicting multiple overlapping hypotheses. Methods with better AP also have better AP^{NO} and likewise improved PQ.

compare to existing metrics like AP and IoU? (3) How do the machine results compare to the human results that we presented previously?

Algorithms and data. We want to understand panoptic segmentation in terms of existing well-established methods. Therefore, we create a basic PS system by applying reasonable heuristics (described shortly) to the output of existing top instance and semantic segmentation systems.

We obtained algorithm output for three datasets. For *Cityscapes*, we use the val set output generated by the current leading algorithms (PSPNet [Zha+17] and Mask R-CNN [He+17] for semantic and instance segmentation, respectively). For *ADE20k*, we received output for the winners of both the semantic [Fu+17; FYM17] and instance [Luo+17; FKM17] segmentation tracks on a 1k subset of test images from the 2017 Places Challenge. For *Vistas*, which is used for the LSUN'17 Segmentation Challenge, the organizers provide us with 1k test images and results from the winning entries for the instance and semantic segmentation tracks [Liu+17a; ZZS17].

Using this data, we start by analyzing PQ for the instance and semantic segmentation tasks separately, and then examine the full panoptic segmentation task. Note that our 'baselines' are very powerful and that simpler baselines may be more reasonable for fair comparison in papers on PS.

Instance segmentation. Instance segmentation algorithms produce overlapping segments. To measure PQ, we must first resolve these overlaps. To do so we develop a simple non-maximum suppression (NMS)-like procedure. We first sort the predicted segments by their confidence scores and remove instances with low scores. Then, we iterate over sorted instances, starting from the most confident. For each instance we first remove pixels which have been assigned to previous segments, then, if a sufficient fraction of the segment remains, we accept the non-overlapping portion, otherwise we discard the entire segment. All thresholds are selected by grid search to optimize PQ. Results on Cityscapes and ADE20k are shown in Table 4.3 (Vistas is omitted as it only had one entry to the 2017 instance challenge). Most importantly, AP and PQ track closely, and we expect improvements in a detector's AP will also improve its PQ.

Cityscapes	IoU	PQ^{St}	SQ^{St}	RQ^{St}
PSPNet multi-scale [Zha+17]	**80.6**	**66.6**	**82.2**	**79.3**
PSPNet single-scale [Zha+17]	79.6	65.2	81.6	78.0
ADE20k	IoU	PQ^{St}	SQ^{St}	RQ^{St}
CASIA_IVA_JD [Fu+17]	**32.3**	**27.4**	**61.9**	**33.7**
G-RMI [FYM17]	30.6	19.3	58.7	24.3

Table 4.4: **Machine results on semantic segmentation** (thing classes ignored). Methods with better mean IoU also show better PQ results. Note that G-RMI has quite low PQ. We found this is because it hallucinates many small patches of classes not present in an image. While this only slightly affects IoU which counts *pixel* errors it severely degrades PQ which counts *instance* errors.

Semantic segmentation. Semantic segmentations have no overlapping segments by design, and therefore we can directly compute PQ. In Table 4.4 we compare mean IoU, a standard metric for this task, to PQ. For Cityscapes, the PQ gap between methods corresponds to the IoU gap. For ADE20k, the gap is much larger. This is because whereas IoU counts correctly predicted pixel, PQ operates at the level of instances. See the Table 4.4 caption for details.

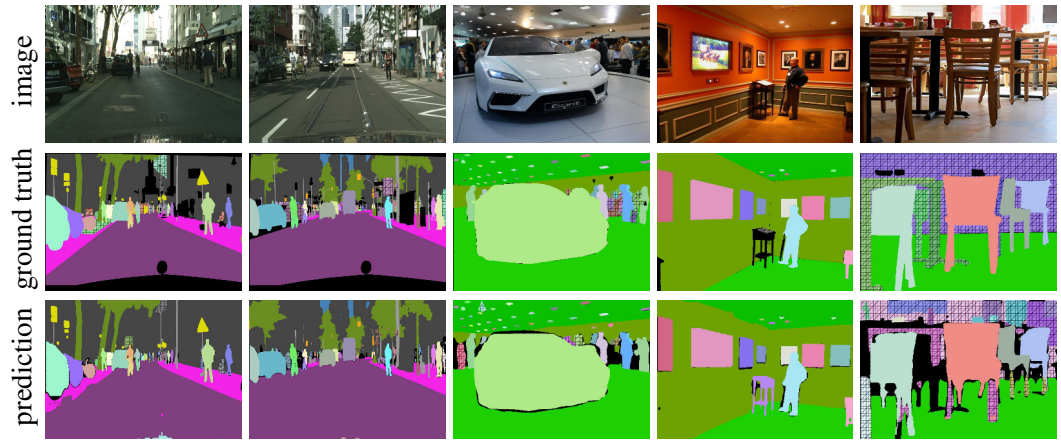


Figure 4.9: **Panoptic segmentation results** on Cityscapes (left two) and ADE20k (right three). Predictions are based on the merged outputs of state-of-the-art instance and semantic segmentation algorithms (see Tables 4.3 and 4.4). Colors for matched segments (IoU>0.5) match (crosshatch pattern indicates unmatched regions and black indicates unlabeled regions). Best viewed in color and with zoom.

Panoptic segmentation. To produce algorithm outputs for PS, we start from the non-overlapping instance segments from the NMS-like procedure described previously. Then, we combine those segments with semantic segmentation results by resolving any overlap between thing and stuff classes in favor of the thing class (*i.e.*, a pixel with a thing and stuff label is assigned the thing label and its instance id). This heuristic is imperfect but sufficient as a baseline.

Table 4.5 compares PQ^{St} and PQ^{Th} computed on the combined ('panoptic') results

Cityscapes	PQ	PQ^{St}	PQ^{Th}
machine-separate	n/a	66.6	54.1
machine-panoptic	61.2	66.4	54.1
ADE20k	PQ	PQ^{St}	PQ^{Th}
machine-separate	n/a	27.4	41.1
machine-panoptic	35.6	24.5	41.1
Vistas	PQ	PQ^{St}	PQ^{Th}
machine-separate	n/a	43.7	35.7
machine-panoptic	38.3	41.8	35.7

Table 4.5: **Panoptic *vs*. independent predictions.** The 'machine-separate' rows show PQ of semantic and instance segmentation methods computed independently (see also Tables 4.3 and 4.4). For 'machine-panoptic', we merge the non-overlapping thing and stuff predictions obtained from state-of-the-art methods into a true panoptic segmentation of the image. Due to the merging heuristic used, PQ^{Th} stays the same while PQ^{St} is slightly degraded.

to the performance achieved from the separate predictions discussed above. For these results we use the winning entries from each respective competition for both the instance and semantic tasks. Since overlaps are resolved in favor of things, PQ^{Th} is constant while PQ^{St} is slightly lower for the panoptic predictions. Visualizations of panoptic outputs are shown in Figure 4.9.

Human *vs*. machine panoptic segmentation. To compare human *vs*. machine PQ, we use the machine panoptic predictions described above. For human results, we use the dual-annotated images described in §4.6 and use bootstrapping to obtain confidence intervals since these image sets are small. These comparisons are imperfect as they use different test images and are averaged over different classes (some classes without matches in the dual-annotated tests sets are omitted), but they can still give some useful signal.

We present the comparison in Table 4.6. For SQ, machines trail humans only slightly. On the other hand, machine RQ is dramatically lower than human RQ, especially on ADE20k and Vistas. This implies that recognition, *i.e*., classification, is the main challenge for current methods. Overall, there is a significant gap between human and machine performance. We hope that this gap will inspire future research for the proposed panoptic segmentation task.

4.8 Future of Panoptic Segmentation

Our goal is to drive research in novel directions by inviting the community to explore the new panoptic segmentation task. We believe that the proposed task can lead to expected and unexpected innovations. We conclude by discussing some of these possibilities and our future plans.

Motivated by simplicity, the PS 'algorithm' in this work is based on the *heuristic*

Cityscapes	PQ	SQ	RQ	PQ^{St}	PQ^{Th}
human	$69.6^{+2.5}_{-2.7}$	$84.1^{+0.8}_{-0.8}$	$82.0^{+2.7}_{-2.9}$	$71.2^{+2.3}_{-2.5}$	$67.4^{+4.6}_{-4.9}$
machine	61.2	81.0	74.4	66.4	54.1
ADE20k	PQ	SQ	RQ	PQ^{St}	PQ^{Th}
human	$67.6^{+2.0}_{-2.0}$	$85.7^{+0.6}_{-0.6}$	$78.6^{+2.1}_{-2.1}$	$71.0^{+3.7}_{-3.2}$	$66.4^{+2.3}_{-2.4}$
machine	35.6	74.4	43.2	24.5	41.1
Vistas	PQ	SQ	RQ	PQ^{St}	PQ^{Th}
human	$57.7^{+1.9}_{-2.0}$	$79.7^{+0.8}_{-0.7}$	$71.6^{+2.2}_{-2.3}$	$62.7^{+2.8}_{-2.8}$	$53.6^{+2.7}_{-2.8}$
machine	38.3	73.6	47.7	41.8	35.7

Table 4.6: **Human *vs*. machine performance.** On each of the considered datasets human performance is much higher than machine performance (approximate comparison, see text for details). This is especially true for RQ, while SQ is closer. The gap is largest on ADE20k and smallest on Cityscapes. Note that as only a small set of human annotations is available, we use bootstrapping and show the the 5^{th} and 95^{th} percentiles error ranges for human results.

combination of outputs from top-performing instance and semantic segmentation systems. This approach is a basic first step, but we expect more interesting algorithms to be introduced. Specifically, we hope to see PS drive innovation in at least two areas: (1) Deeply integrated end-to-end models that simultaneously address the dual stuff-and-thing nature of PS. A number of instance segmentation approaches including [Liu+17b; AT17; BU17; Kir+17] are designed to produce non-overlapping instance predictions and could serve as the foundation of such a system. (2) Since a PS cannot have overlapping segments, some form of higher-level 'reasoning' may be beneficial, for example, based on extending learnable NMS [DRF11; HBS17; Hu+18] to PS. We hope that the panoptic segmentation task will invigorate research in these areas leading to exciting new breakthroughs in vision.

Finally, we are working with competition organizers to extend popular segmentation datasets to include a panoptic segmentation track. Currently the COCO [Lin+14], Vistas [Neu+17], and ADE20k [Zho+17] challenges are considering featuring a panoptic segmentation track in 2018. We hope this will lead to a broad adoption of the proposed joint task.

Chapter 5

Discussion

In this thesis we explored three different aspects of image segmentation:

- We proposed a novel formulation for the problem of producing multiple diverse solutions for a single input image;
- We presented a new bottom-up approach that infers instance segmentation using global reasoning;
- We introduced the panoptic segmentation task accompanied by a panoptic quality metric as new, rich, and coherent segmentation task.

We hope that these contributions will facilitate future research of robust and effective scene understanding systems.

Scene understanding perspective. In our work we address three crucial aspects of image segmentation: *diversity*, *global reasoning* and *general segmentation formulation*. These are essential ingredients for future segmentation systems that can be used in real-world scene understanding applications. The explicit incorporation of the notion of diversity makes it tolerant to the ambiguity of natural tasks. Moreover, it helps to overcome possible shortage of training data. Global reasoning is another aspect that makes the final system more robust. Joint inference of all segments provides the ability to use high-level knowledge. Even with the lack of direct clues, a correct local decision can be made based on the scene structure.

The panoptic formulation combines previously distinct semantic and instance segmentation tasks. Both are essential for various real-world vision applications. The method that brings them together produces coherent output and resolves possible inconsistencies. Future processing, therefore, can rely on consistent information as opposed to two potentially conflicting input sources. This makes the use of segmentation techniques easier for high-level applications.

5.1 Limitations and Future Work

While we made some progress in several aspects of image segmentation, there are still some limitations and open research questions left to be addressed. In this section we discuss these limitations in detail.

5.1.1 Multiple Diverse Solutions

In this work we presented a novel problem formulation that is capable of producing multiple diverse solutions using a single trained model that originally outputs only a single solution. This formulation generalizes previous methods. We proposed several approximate and exact inference techniques that are efficient for certain models (pair-wise or submodular) and diversity measures (node-wise measures). Together with previously known techniques they form a set of tools that can satisfy different quality/efficiency trade-offs. In this section we discuss some limitations of our approach and outline future research directions for diversity methods.

Learning the diversity measure. In our work we assume diversity measures Δ to be pre-defined. Single parameter λ that sets the trade-off between quality of each solution and their diversity was tuned via grid search. While our framework with standard Hamming distance diversity measure demonstrates strong performance, tuning the diversity measure together with the original model for an application at hand is a very promising direction for future research. The main obstacles for a breakthrough in this area are symmetry of the problem (any permutation of diverse solutions is valid) and high probability of collapsing to just one solution. Despite these difficulties, some work has already been done in this direction [LCK18; Lee+16]. The introduction of new datasets that have multiple ground truth annotations for each image will most likely ease these issues and facilitate this research area.

High order diversity measures. In Chapter 2 we consider only node-wise diversity measures. Node-wise diversity measures may be sufficient if the original model is a CRF with pair-wise or high-order potentials. These potentials ensure that while being diverse different solutions are consistent. However, if the original model is a simple CNN that produces independent predictions for each pixel, node-wise diversity is not helpful. For this case, our formulation splits into small per-pixel problems that may produce inconsistent results. In [PJB14] approximate inference techniques for several high-order diversity measures were proposed. The development of efficient methods for a broad range of high-order diversity measures like the difference in the number of connected components in segmentation or the difference of shapes is a very promising future research direction. The ability to produce sensible diverse solutions from a single CNN will make segmentation systems more robust and potentially will help to interpret the behavior of trained CNNs.

Efficient general solver. In our work we develop the K-Clique Encoding optimization technique. With a Quadratic Pseudo-Bolean Optimization (QPBO) solver [Rot+07] on each step, the technique is applicable to arbitrary pair-wise original CRFs with a node-wise diversity measure. The LP-based approach for diversity inference is likely to be more efficient and applicable to a broader range of models including high-order CRFs. The development of such solver will facilitate the adoption of methods that produce multiple diverse solutions in adjacent science fields like bio-imaging where high-order CRFs are very common.

5.1.2 Bottom-Up Instance Segmentation Framework

In Chapter 3 we presented a novel bottom-up approach for instance segmentation – InstanceCut. This method is a very straightforward implementation of the bottom-up paradigm. It infers segmentation for all instances globally based on local clues from two Fully Convolutional Networks. While promising results were shown on a challenging dataset, InstanceCut has some downsides that could be addressed in the future. In fact, since the method was published, novel bottom-up approaches partially addressing these issues have already appeared. In what follows we discuss these issues in detail.

(a) InstanceCut Prediction (b) Ground Truth

Figure 5.1: Left car is occluded by the person in front. InstanceCut identifies two split parts of the car as independent car instances.

Grouping of connected components. By design InstanceCut inference is not able to recognize instances split by occlusion for several connected components. Instead it recognizes each instance as a separate instance, see Fig. 5.1 for illustration. Note, however, that each connected component segmentation is fine-grained. Hence, instances that split into several connected components can be recovered via some post-processing scheme. In fact, recent instance segmentation work [Liu+17b] that follows the bottom-up paradigm has shown that such grouping is quite effective. Making the grouping step a part of the whole training procedure is an interesting direction for future work.

End-to-end training. In InstanceCut two FCNs were trained independently to produce per-pixel scores of semantic labels and per-pixel probabilities of instance edges respectively. A unified end-to-end training technique that trains both FCNs together towards the final goal of great instance segmentation performance is a promising direction for future research. In fact, currently neither top-down based methods nor bottom-up methods are fully end-to-end trainable. State-of-the-art top-down approaches like Mask R-CNN [He+17] and the Path Aggregation Network [Liu+18] use Non-Maximal Suppression (NMS) to filter out duplicates. Recent bottom-up approaches for instance segmentation [Liu+17b; BU17; DBNVG17] use a pipeline of neural networks with heuristics on top to produce the final output. As a first step, recent work proposes a fully end-to-end trainable system [Sal+17] based on recurrent neural networks. Some work has been done to make NMS trainable [Hu+18] and to train it together with the whole system. We expect more progress in this direction by the computer vision community in the next few years.

Hybrid approach. InstanceCut and more recent bottom-up instance segmentation frameworks have shown great performance being able to segment out heavily occluded

objects based on local information like object boundaries. These methods are able to segment such objects even when state-of-the-art recognition approaches (the backbone of top-down approaches) fail to recognize them. At the same time, due to the usage of strong recognition sub-networks, top-down based methods are able to segment out very small and distant objects with very little visual information. These advantages of the two paradigms are, in fact, complementary. This observation is backed by evaluation metrics. Mask R-CNN [He+17] (strong top-down approach) shows 26.2 overall average precision (AP) and 40.1 AP-50m (AP for objects that are not further than 50 meters from the camera). At the same time the state-of-the-art bottom-up approach Sequential Grouping Network [Liu+17b] demonstrates lower overall AP 25.0, showing significantly better performance for close objects – 44.5 AP-50m. These numbers imply that the bottom-up approach works better in cases of rich visual information and the recognition system in top-down approaches helps them to work better for other objects. Given this observation, development of a hybrid method that combines both bottom-up and top-down paradigms is a very promising direction for future work. Recently, [Che+17b] proposed a hybrid system for fine-grained instance segmentation.

5.1.3 Segmentation for Scene Understanding Applications

In this thesis we introduced Panoptic Segmentation task. This task combines semantic and instance segmentation together into a single consistent segmentation task. The proposed Panoptic Quality metric measures performance for all categories (both things and stuff) in a unified manner. Panoptic Segmentation provides rich and coherent scene information. Taking into account its practical importance, we aim to revive the interest of our community in a more unified view of image segmentation. In this section we discuss some potential ways to further improve and generalize the task.

Ambiguity. The panoptic segmentation format assumes a single prediction for each pixel of an image. While the simplicity of this approach is appealing, it does not take into account all properties of real-world scenes. One of these properties is natural scene ambiguity. Our study of human annotations has shown that expert annotators segment out object masks fairly consistently. At the same time, the exact semantic class of the object is often not clear without any additional context (previous frames or some meta data); different annotators assign different labels for the same object. The panoptic segmentation format can be extended to deal with this natural property of segmentation. Instead of having a single semantic label for each segment, the distribution over possible labels can be predicted. This distribution gives more information to downstream systems where exact labels are determined using additional context, specific constraints, and properties of the task at hand. This behavior is similar to modern instance segmentation methods that provide confidence scores for each mask [He+17; Liu+18].

Amodal panoptic segmentation. In our work on panoptic segmentation we use datasets that mark only visible parts of objects, *i.e.* there are no occlusions in the annotations. We enforce no occlusions in the output format respectively. Promising future research is to extend the panoptic task to amodal segmentation setups. Amodal datasets like [Zhu+17] annotate objects to their full extent, not only visible parts.

Holistic Scene Understanding. In this thesis we focus on the segmentation part of the scene understanding problem. While the panoptic segmentation task already provides richer information about the scene than any one of the tasks it unifies, we hope that the panoptic task will evolve further by incorporating modalities beyond segmentation annotations, *e.g*. adding depth information, key-point, optical flow, etc. The resulting general scene understanding task with a new unified quality metric will help to combine stand-alone tasks in a more conscious way than multi-task approach. We hope that this evolution will lead to synergy effects between different modalities and will make holistic scene understanding possible.

Bibliography

[Ach+12] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. "SLIC superpixels compared to state-of-the-art superpixel methods". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 34.11 (2012), pp. 2274–2282.

[Ade01] E. H. Adelson. "On seeing stuff: the perception of materials by humans and machines". In: *Human Vision and Electronic Imaging*. 2001.

[Arb+11] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. "Contour detection and hierarchical image segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 33.5 (2011), pp. 898–916.

[Arb+14] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. "Multiscale combinatorial grouping". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014, pp. 328–335.

[AT17] A. Arnab and P. H. Torr. "Pixelwise instance segmentation with a dynamically instantiated network". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[Aro+15] C. Arora, S. Banerjee, P. Kalra, and S. Maheshwari. "Generalized Flows for Optimal Inference in Higher Order MRF-MAP". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2015).

[Bac13] F. Bach. "Learning with Submodular Functions: A Convex Optimization Perspective". In: *Foundations and Trends in Machine Learning* 6.2-3 (2013), pp. 145–373.

[BU17] M. Bai and R. Urtasun. "Deep watershed transform for instance segmentation". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[BBC04] N. Bansal, A. Blum, and S. Chawla. "Correlation Clustering". In: *Machine Learning* 56.1 (2004), pp. 89–113.

[Bat12] D. Batra. "An efficient message-passing algorithm for the M-best MAP problem". In: *Conference on Uncertainty in Artificial Intelligence (UAI)*. 2012.

[Bat+12] D. Batra, P. Yadollahpour, A. Guzman-Rivera, and G. Shakhnarovich. "Diverse M-Best Solutions in Markov Random Fields". In: *European Conference on Computer Vision (ECCV)*. Springer Berlin/Heidelberg, 2012.

[Bei+14] T. Beier, T. Kroeger, J. H. Kappes, U. Köthe, and F. A. Hamprecht. "Cut, Glue, & Cut: A Fast, Approximate Solver for Multicut Partitioning". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2014, pp. 73–80.

[BST15a] G. Bertasius, J. Shi, and L. Torresani. “Deepedge: A multi-scale bifurcated deep network for top-down contour detection”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 4380–4389.

[BST15b] G. Bertasius, J. Shi, and L. Torresani. “High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 504–512.

[BST16] G. Bertasius, J. Shi, and L. Torresani. “Semantic segmentation with boundary neural fields”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[BZ87] A. Blake and A. Zisserman. *Visual reconstruction*. MIT press, 1987.

[BJ01] Y. Boykov and M.-P. Jolly. “Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2001.

[BK04] Y. Boykov and V. Kolmogorov. “An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 26.9 (2004), pp. 1124–1137.

[BVZ01] Y. Boykov, O. Veksler, and R. Zabih. “Fast approximate energy minimization via graph cuts”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2001).

[CUF18] H. Caesar, J. Uijlings, and V. Ferrari. “COCO-Stuff: Thing and Stuff Classes in Context”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[Can86] J. Canny. “A computational approach to edge detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 6 (1986), pp. 679–698.

[Cao+17] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. “Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[Car+12] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. “Semantic segmentation with second-order pooling”. In: *European Conference on Computer Vision (ECCV)*. Springer. 2012, pp. 430–443.

[Cha05] A. Chambolle. “Total variation minimization and a class of binary MRF models”. In: *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*. Springer. 2005, pp. 136–152.

[CE05] T. F. Chan and S. Esedoglu. “Aspects of Total Variation Regularized L1 Function Approximation”. In: *SIAM Journal on Applied Mathematics* 65.5 (2005), pp. 1817–1837.

[Che+13] C. Chen, V. Kolmogorov, Y. Zhu, D. N. Metaxas, and C. H. Lampert. “Computing the M Most Probable Modes of a Graphical Model”. In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2013.

[Che+17a] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2017).

[Che+17b] L.-C. Chen, A. Hermans, G. Papandreou, F. Schroff, P. Wang, and H. Adam. "MaskLab: Instance Segmentation by Refining Object Detection with Semantic and Direction Features". In: *arXiv preprint arXiv:1712.04837* (2017).

[CLY15] Y.-T. Chen, X. Liu, and M.-H. Yang. "Multi-instance object segmentation with occlusion handling". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3470–3478.

[CR93] S. Chopra and M. R. Rao. "The partition problem". In: *Mathematical Programming* 59.1 (1993), pp. 87–115.

[Cor+16] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. "The Cityscapes Dataset for Semantic Urban Scene Understanding". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Cor09] T. H. Cormen. *Introduction to algorithms*. MIT press, 2009.

[DHS15] J. Dai, K. He, and J. Sun. "Convolutional feature masking for joint object and stuff segmentation". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3992–4000.

[DHS16] J. Dai, K. He, and J. Sun. "Instance-aware semantic segmentation via multi-task network cascades". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[Dai+16] J. Dai, K. He, Y. Li, S. Ren, and J. Sun. "Instance-sensitive fully convolutional networks". In: *European Conference on Computer Vision (ECCV)* (2016).

[DS04] J. Darbon and M. Sigelle. "Exact optimization of discrete constrained total variation minimization problems". In: *International Workshop on Combinatorial Image Analysis*. Springer. 2004, pp. 548–557.

[DBNVG17] B. De Brabandere, D. Neven, and L. Van Gool. "Semantic instance segmentation with a discriminative loss function". In: *arXiv preprint arXiv:1708.02551* (2017).

[DRF11] C. Desai, D. Ramanan, and C. C. Fowlkes. "Discriminative models for multi-class object layout". In: *International Journal of Computer Vision (IJCV)* (2011).

[Dol+12] P. Dollár, C. Wojek, B. Schiele, and P. Perona. "Pedestrian Detection: An Evaluation of the State of the Art". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2012).

[DZ15] P. Dollár and C. L. Zitnick. "Fast edge detection using structured forests". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 37.8 (2015), pp. 1558–1570.

[Eve+15] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. "The PASCAL visual object classes challenge: A retrospective". In: *International Journal of Computer Vision (IJCV)* (2015).

[FKM17] A. Fathi, N. Kanazawa, and K. Murphy. *Places Challenge 2017: instance segmentation, G-RMI team*. 2017.

[FYM17] A. Fathi, K. Yang, and K. Murphy. *Places Challenge 2017: scene parsing, G-RMI team*. 2017.

[Fix+11] A. Fix, A. Gruber, E. Boros, and R. Zabih. "A graph cut algorithm for higher-order Markov random fields". In: *The IEEE International Conference on Computer Vision (ICCV)*. 2011.

[FI03] L. Fleischer and S. Iwata. “A push-relabel framework for submodular function minimization and applications to parametric optimization”. In: *Discrete Applied Mathematics* 131.2 (2003), pp. 311–322.

[FS08] V. Franc and B. Savchynskyy. “Discriminative learning of max-sum classifiers”. In: *Journal of Machine Learning Research (JMLR)* 9 (2008), pp. 67–104.

[FG09] M. Fromer and A. Globerson. “An LP View of the M-best MAP problem”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2009.

[Fu+17] J. Fu, J. Liu, L. Guo, H. Tian, F. Liu, H. Lu, Y. Li, Y. Bao, and W. Yan. *Places Challenge 2017: scene parsing, CASIA_IVA_JD team*. 2017.

[GGT89] G. Gallo, M. D. Grigoriadis, and R. E. Tarjan. “A fast parametric maximum flow algorithm and applications”. In: *SIAM Journal on Computing* 18.1 (1989), pp. 30–55.

[GL14] Y. Ganin and V. Lempitsky. “Nˆ 4-Fields: Neural Network Nearest Neighbor Fields for Image Transforms”. In: *Asian Conference on Computer Vision (ACCV)*. Springer. 2014, pp. 536–551.

[GG84] S. Geman and D. Geman. “Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 6 (1984), pp. 721–741.

[GF16] G. Ghiasi and C. C. Fowlkes. “Laplacian Pyramid Reconstruction and Refinement for Semantic Segmentation”. In: *European Conference on Computer Vision (ECCV)*. Springer. 2016, pp. 519–534.

[GRBK12] A. Guzman-Rivera, D. Batra, and P. Kohli. “Multiple Choice Learning: Learning to Produce Multiple Structured Outputs”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2012.

[GRKB13] A. Guzman-Rivera, P. Kohli, and D. Batra. “DivMCuts: Faster Training of Structural SVMs with Diverse M-Best Cutting-Planes”. In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2013.

[Guz+14] A. Guzman-Rivera, P. Kohli, D. Batra, and R. A. Rutenbar. “Efficiently Enforcing Diversity in Multi-Output Structured Prediction”. In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2014.

[Ham14] F. A. Hamprecht. “Asymmetric Cuts: Joint Image Labeling and Partitioning”. In: *German Conference Pattern Recognition (GCPR)*. Vol. 8753. Springer. 2014, p. 199.

[HS85] R. M. Haralick and L. G. Shapiro. “Image segmentation techniques”. In: *Computer Vision, Graphics, and Image Processing* 29.1 (1985), pp. 100–132.

[Har+14] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. “Simultaneous detection and segmentation”. In: *European Conference on Computer Vision (ECCV)*. Springer. 2014, pp. 297–312.

[Har+15] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. “Hypercolumns for object segmentation and fine-grained localization”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 447–456.

[He+17] K. He, G. Gkioxari, P. Dollár, and R. Girshick. “Mask R-CNN”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2017.

[Hoc01] D. S. Hochbaum. “An efficient algorithm for image segmentation, Markov random fields and related problems”. In: *Journal of the ACM (JACM)* 48.4 (2001), pp. 686–701.

[Hoc08] D. S. Hochbaum. “The pseudoflow algorithm: A new algorithm for the maximum-flow problem”. In: *Operations research* 56.4 (2008), pp. 992–1009.

[Hoc13] D. S. Hochbaum. “Multi-Label Markov Random Fields as an Efficient and Effective Tool for Image Segmentation, Total Variations and Regularization”. In: *Numerical Mathematics: Theory, Methods and Applications* 6 (01 Feb. 2013), pp. 169–198.

[HS97] S. Hochreiter and J. Schmidhuber. “Long short-term memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780.

[HBS17] J Hosang, R Benenson, and B Schiele. “Learning Non-maximum Suppression”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2017).

[Hos+15] J. Hosang, R. Benenson, P. Dollár, and B. Schiele. “What makes for effective detection proposals?” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2015).

[Hu+18] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei. “Relation Networks for Object Detection”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[HL15] J.-J. Hwang and T.-L. Liu. “Pixel-wise deep learning for contour detection”. In: *arXiv preprint arXiv:1504.01989* (2015).

[Ish03] H. Ishikawa. “Exact optimization for Markov random fields with convex priors”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2003).

[Iso+14] P. Isola, D. Zoran, D. Krishnan, and E. H. Adelson. “Crisp boundary detection using pointwise mutual information”. In: *European Conference on Computer Vision (ECCV)*. Springer. 2014, pp. 799–814.

[Kap+15] J. H. Kappes et al. “A Comparative Study of Modern Inference Techniques for Structured Discrete Energy Minimization Problems”. English. In: *International Journal of Computer Vision (IJCV)* (2015), pp. 1–30.

[KL70] B. W. Kernighan and S. Lin. “An efficient heuristic procedure for partitioning graphs”. In: *Bell System Technical Journal* 49.2 (1970), pp. 291–307.

[Keu+15] M. Keuper, E. Levinkov, N. Bonneel, G Lavou, T. Brox, and B. Andres. “Efficient decomposition of image and mesh graphs by lifted multicuts”. In: *The IEEE International Conference on Computer Vision (ICCV)*. IEEE. 2015, pp. 1751–1759.

[Kir+15a] A. Kirillov, B. Savchynskyy, D. Schlesinger, D. Vetrov, and C. Rother. “Inferring M-Best Diverse Labelings in a Single One”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2015.

[Kir+15b] A. Kirillov, D. Schlesinger, D. P. Vetrov, C. Rother, and B. Savchynskyy. “M-Best-Diverse Labelings for Submodular Energies and Beyond”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2015.

[Kir+16] A. Kirillov, A. Shekhovtsov, C. Rother, and B. Savchynskyy. “Joint M-Best-Diverse Labelings as a Parametric Submodular Minimization”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2016.

[Kir+17] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. “InstanceCut: from edges to instances with multicut”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[Kiv+14] J. J. Kivinen, C. K. Williams, N. Heess, and D. Technologies. “Visual Boundary Prediction: A Deep Neural Prediction Network and Quality Dissection.” In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. Vol. 1. 2. 2014, p. 9.

[KT07] P. Kohli and P. H. Torr. “Dynamic graph cuts for efficient inference in Markov random fields”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2007).

[Kok17] I. Kokkinos. “UberNet: Training a Universal Convolutional Neural Network for Low-, Mid-, and High-Level Vision using Diverse Datasets and Limited Memory”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[Kol12] V. Kolmogorov. “Minimizing a sum of submodular functions”. In: *Discrete Applied Mathematics* (2012).

[KZ04] V. Kolmogorov and R. Zabih. “What energy functions can be minimized via graph cuts?” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2004).

[Kol11] V. Koltun. “Efficient inference in fully connected crfs with gaussian edge potentials”. In: *Advances in Neural Information Processing Systems (NIPS)* (2011).

[KSH12] A. Krizhevsky, I. Sutskever, and G. Hinton. “ImageNet classification with deep convolutional neural networks”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2012.

[KT10] A. Kulesza and B. Taskar. “Structured Determinantal Point Processes”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2010.

[Law72] E. L. Lawler. “A Procedure for Computing the K Best Solutions to Discrete Optimization Problems and Its Application to the Shortest Path Problem”. In: *Management Science* 18.7 (1972).

[LJK17] S.-H. Lee, W.-D. Jang, and C.-S. Kim. “Temporal Superpixels Based on Proximity-Weighted Patch Matching”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2017.

[Lee+16] S. Lee, S. P. S. Prakash, M. Cogswell, V. Ranjan, D. Crandall, and D. Batra. “Stochastic multiple choice learning for training diverse deep ensembles”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2016, pp. 2119–2127.

[Lev+17] E. Levinkov, J. Uhrig, S. Tang, M. Omran, E. Insafutdinov, A. Kirillov, C. Rother, T. Brox, B. Schiele, and B. Andres. “Joint Graph Decomposition & Node Labeling: Problem, Algorithms, Applications”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

[Li+17] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. “Fully convolutional instance-aware semantic segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[LCK18] Z. Li, Q. Chen, and V. Koltun. “Interactive Image Segmentation with Latent Diversity”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[Lia+16] X. Liang, Y. Wei, X. Shen, Z. Jie, J. Feng, L. Lin, and S. Yan. “Reversible recursive instance-level object segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[Lia+17] X. Liang, Y. Wei, X. Shen, J. Yang, L. Lin, and S. Yan. “Proposal-free network for instance-level object segmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2017).

[LZD13] J. J. Lim, C. L. Zitnick, and P. Dollár. “Sketch tokens: A learned mid-level representation for contour and object detection”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013, pp. 3158–3165.

[LSR+16] G. Lin, C. Shen, I. Reid, et al. “Efficient piecewise training of deep structured models for semantic segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[Lin+14] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. “Microsoft coco: Common objects in context”. In: *European Conference on Computer Vision (ECCV)*. Springer. 2014, pp. 740–755.

[LYT11] C. Liu, J. Yuen, and A. Torralba. “SIFT flow: Dense correspondence across scenes and its applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2011).

[Liu+17a] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. *LSUN’17: insatnce segmentation task, UCenter winner team*. 2017.

[Liu+16] S. Liu, X. Qi, J. Shi, H. Zhang, and J. Jia. “Multi-scale Patch Aggregation (MPA) for Simultaneous Detection and Segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 3141–3149.

[Liu+17b] S. Liu, J. Jia, S. Fidler, and R. Urtasun. “SGN: Sequential Grouping Networks for Instance Segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[Liu+18] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. “Path aggregation network for instance segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[Liu+15] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang. “Semantic image segmentation via deep parsing network”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1377–1385.

[LSD15] J. Long, E. Shelhamer, and T. Darrell. “Fully convolutional networks for semantic segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3431–3440.

[Luo+17] R. Luo, B. Jiang, T. Xiao, C. Peng, Y. Jiang, Z. Li, X. Zhang, G. Yu, Y. Mu, and J. Sun. *Places Challenge 2017: instance segmentation, Megvii (Face++) team*. 2017.

[Mal+16] J. Malik, P. Arbeláez, J. Carreira, K. Fragkiadaki, R. Girshick, G. Gkioxari, S. Gupta, B. Hariharan, A. Kar, and S. Tulsiani. "The three R's of computer vision: Recognition, reconstruction and reorganization". In: *Journal of Pattern Recognition Letters (PRL)* (2016).

[Mar82] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., 1982.

[MFM04] D. R. Martin, C. C. Fowlkes, and J. Malik. "Learning to detect natural image boundaries using local brightness, color, and texture cues". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2004).

[Men+14] B. Menze et al. "The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)". In: *IEEE Transactions on Medical Imaging* (2014), p. 33.

[Mis+16] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert. "Cross-stitch networks for multi-task learning". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[Neu+17] G. Neuhold, T. Ollmann, S. Rota Bulò, and P. Kontschieder. "The mapillary vistas dataset for semantic understanding of street scenes". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[Nil98] D. Nilsson. "An efficient algorithm for finding the M most probable configurationsin probabilistic expert systems". In: *Statistics and Computing* 8.2 (1998), pp. 159–173.

[PY11] G. Papandreou and A. Yuille. "Perturb-and-MAP random fields: Using discrete optimization to learn and sample from energy models". In: *The IEEE International Conference on Computer Vision (ICCV)*. 2011.

[PCD15] P. O. Pinheiro, R. Collobert, and P. Dollár. "Learning to segment object candidates". In: *Advances in Neural Information Processing Systems (NIPS)*. 2015.

[Pin+16] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. "Learning to refine object segments". In: *European Conference on Computer Vision (ECCV)*. 2016.

[PZ11] J. Porway and S.-C. Zhu. "C^ 4: Exploring Multiple Solutions in Graphical Models by Cluster Sampling". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 33.9 (2011), pp. 1713–1727.

[PJB14] A. Prasad, S. Jegelka, and D. Batra. "Submodular meets Structured: Finding Diverse Subsets in Exponentially-Large Structured Item Sets". In: *Advances in Neural Information Processing Systems (NIPS)*. 2014.

[PTB14] V. Premachandran, D. Tarlow, and D. Batra. "Empirical Minimum Bayes Risk Prediction: How to extract an extra few % performance from vision models with just three more parameters". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014.

[RB12] V. Ramakrishna and D. Batra. "Mode-Marginals: Expressing Uncertainty via Diverse M-Best Solutions". In: *NIPS Workshop on Perturbations, Optimization, and Statistics*. 2012.

[RZ16] M. Ren and R. S. Zemel. "End-to-End Instance Segmentation and Counting with Recurrent Attention". In: *arXiv preprint arXiv:1605.09410* (2016).

[Ren+15] S. Ren, K. He, R. Girshick, and J. Sun. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". In: *Advances in Neural Information Processing Systems (NIPS)*. 2015.

[RPT16] B. Romera-Paredes and P. H. Torr. “Recurrent instance segmentation”. In: *European Conference on Computer Vision (ECCV)* (2016).

[RFB15] O. Ronneberger, P. Fischer, and T. Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer. 2015, pp. 234–241.

[RKB04] C. Rother, V. Kolmogorov, and A. Blake. “Grabcut: Interactive foreground extraction using iterated graph cuts”. In: *ACM Transactions on Graphics (TOG)*. Vol. 23. 3. ACM. 2004, pp. 309–314.

[Rot+07] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer. “Optimizing binary MRFs via extended roof duality”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2007, pp. 1–8.

[Rus+15] O. Russakovsky et al. “ImageNet Large Scale Visual Recognition Challenge”. In: *International Journal of Computer Vision (IJCV)* (2015).

[Sal+17] A. Salvador, M. Bellver, M. Baradad, F. Marqués, J. Torres, and X. Giro-i Nieto. “Recurrent Neural Networks for Semantic Instance Segmentation”. In: *arXiv preprint arXiv:1712.00617* (2017).

[SF06] D. Schlesinger and B. Flach. *Transforming an arbitrary minsum problem into a binary one*. TU Dresden, Fak. Informatik, 2006.

[SH02] M. I. Schlesinger and V. Hlavac. *Ten lectures on statistical and structural pattern recognition*. 2002.

[SU15] A. G. Schwing and R. Urtasun. “Fully connected deep structured networks”. In: *arXiv preprint arXiv:1503.02351* (2015).

[She+15] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang. “Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3982–3991.

[SM00] J. Shi and J. Malik. “Normalized cuts and image segmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 22.8 (2000), pp. 888–905.

[Sho+06] J. Shotton, J. Winn, C. Rother, and A. Criminisi. “Textonboost: Joint appearance, shape and context modeling for multi-class object recog. and segm.” In: *European Conference on Computer Vision (ECCV)*. 2006.

[Sun+14] M. Sun, B. Kim, P. Kohli, and S. Savarese. “Relating things and stuff via object property interactions”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2014).

[Sze+08] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. “A comparative study of energy minimization methods for markov random fields with smoothness-based priors”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 30.6 (2008), pp. 1068–1080.

[TGZ10] D. Tarlow, I. E. Givoni, and R. S. Zemel. “HOP-MAP: Efficient message passing with high order potentials”. In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2010.

[TL13] J. Tighe and S. Lazebnik. “Finding things: Image parsing with regions and per-exemplar detectors”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013.

[TNL14] J. Tighe, M. Niethammer, and S. Lazebnik. “Scene parsing with object instances and occlusion ordering”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014.

[Top78] D. M. Topkis. “Minimizing a submodular function on a lattice”. In: *Operations research* 26.2 (1978), pp. 305–321.

[TZ02] Z. Tu and S.-C. Zhu. “Image segmentation by data-driven Markov chain Monte Carlo”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 24.5 (2002), pp. 657–673.

[Tu+05] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu. “Image parsing: Unifying segmentation, detection, and recognition”. In: *International Journal of Computer Vision (IJCV)* (2005).

[Uhr+16] J. Uhrig, M. Cordts, U. Franke, and T. Brox. “Pixel-level encoding and depth layering for instance-level semantic labeling”. In: *German Conference Pattern Recognition (GCPR)* (2016).

[VML94] R. Vaillant, C. Monrocq, and Y. LeCun. “Original approach for the localisation of objects in images”. In: *IEE Proceedings - Vision, Image and Signal Processing* (1994).

[VR79] C. Van Rijsbergen. *Information Retrieval*. London: Butterworths, 1979.

[VS91] L. Vincent and P. Soille. “Watersheds in digital spaces: an efficient algorithm based on immersion simulations”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 13.6 (1991), pp. 583–598.

[VJ01] P. Viola and M. Jones. “Rapid object detection using a boosted cascade of simple features”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2001.

[WJ08] M. J. Wainwright and M. I. Jordan. “Graphical models, exponential families, and variational inference”. In: *Foundations and Trends in Machine Learning* (2008).

[Wer07] T. Werner. “A Linear Programming Approach to Max-sum Problem: A Review”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 29.7 (2007).

[Wer23] M. Wertheimer. “Laws of organization in perceptual forms”. In: *A source book of Gestalt Psychology* (1923).

[Wes01] D. B. West. *Introduction to Graph Theory*. Vol. 2. Prentice hall Upper Saddle River, 2001.

[WSH16] Z. Wu, C. Shen, and A. v. d. Hengel. “Bridging Category-level and Instance-level Semantic Image Segmentation”. In: *arXiv preprint arXiv:1605.06885* (2016).

[XT15] S. Xie and Z. Tu. “Holistically-nested edge detection”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1395–1403.

[YBS13] P. Yadollahpour, D. Batra, and G. Shakhnarovich. “Discriminative Re-ranking of Diverse Segmentations”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013.

[Yan+12] Y. Yang, S. Hallman, D. Ramanan, and C. C. Fowlkes. “Layered object models for image segmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2012).

[YW04] C. Yanover and Y. Weiss. “Finding the M most probable configurations using loopy belief propagation”. In: *Advances in Neural Information Processing Systems (NIPS)*. 2004.

[YFU12] J. Yao, S. Fidler, and R. Urtasun. “Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2012.

[YK16] F. Yu and V. Koltun. “Multi-Scale Context Aggregation by Dilated Convolutions”. In: *International Conference on Learning Representations (ICLR)*. 2016.

[Zag+16] S. Zagoruyko, A. Lerer, T.-Y. Lin, P. O. Pinheiro, S. Gross, S. Chintala, and P. Dollár. “A MultiPath Network for Object Detection”. In: *The British Machine Vision Conference (BMVC)* (2016).

[ZZS17] Y. Zhang, H. Zhao, and J. Shi. *LSUN’17: semantic segmentation task, PSPNet winner team*. 2017.

[ZFU16] Z. Zhang, S. Fidler, and R. Urtasun. “Instance-level segmentation with deep densely connected MRFs”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[Zha+15] Z. Zhang, A. G. Schwing, S. Fidler, and R. Urtasun. “Monocular object instance segmentation and depth ordering with cnns”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 2614–2622.

[Zha+17] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. “Pyramid Scene Parsing Network”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[Zhe+15] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr. “Conditional random fields as recurrent neural networks”. In: *The IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1529–1537.

[Zho+17] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. “Scene Parsing through ADE20K Dataset”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[Zhu+17] Y. Zhu, Y. Tian, D. Mexatas, and P. Dollár. “Semantic amodal segmentation”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.