Towards Efficient Deep Learning for Computer Vision

Sudhanshu Mittal

Dissertation zur Erlangung des Doktorgrades der Technischen Fakultät der Albert-Ludwigs-Universität Freiburg

Datum der mündlichen Prüfung:	01.09.2023
Dekan:	Prof. Dr. Frank Balle
Erstgutachter und Betereuer:	Prof. Dr. Thomas Brox
Zweitgutachter:	Prof. Dr. Carsten Rother
Beisitzer:	Prof. Dr. Abhinav Valada
Vorsitzer:	Prof. Dr. Matthias Teschner

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text.

Sudhanshu Mittal

Acknowledgements

Throughout my journey, I have had immense support and assistance from a lot of wonderful people, and I take this opportunity to thank them. I would like to thank everyone in my family. Their constant trust and support is invaluable and have helped me tremendously throughout this endeavor. I owe immense gratitude to my Ph.D. advisor Prof. Thomas Brox. I thank him for giving me an opportunity to work at the Computer Vision Group. I have learned a lot from him about both scientific and non-scientific things. I thank him for his regular support and encouragement, which has hopefully made me a better researcher. I would like to thank especially Maxim Tatarchenko for being an amazing mentor and co-author and for inspiring me to do good research. His clarity of scientific thinking is admirable. I would like to thank Stefan Teister, Nikolaus Mayer, and Philipp Schröppel for providing amazing IT support, and Jasmin Anders and Julia Willi for all the administrative help. I would also like to thank my previous advisors, Prof. K. Madhava Krishna and Prof. Avinash Gautam, who introduced me to the field of Machine Learning.

I would like to thank my other co-authors, Özgün Çiçek, Silvio Galesso, Maria Bravo, Joshua Niemeijer, and Simon Ging, for their collaboration and scientific discussions. I would like to thank Maria Bravo for being an amazing friend and colleague. Her support was crucial in the second half of my Ph.D. journey. I also thank her for introducing me to vision-language models and collaborating on some really interesting research work. I would like to thank Ayush Dewan for being a good friend and a strong ally. His support was crucial for my stay in Freiburg. I would like to thank Philipp Schröppel, Silvio Galesso, and Maria Bravo for proofreading this thesis, and Simon Ging for helping with writing the German version of the abstract. Sports has been an important part during my Ph.D. I would like to thank my sports buddies Ayush Dewan, Noha Radwann, Andreas Eitel, Jan Bechtold, and Maria Bravo for motivating me to stay fit and healthy. LMB always has been a great office place to work. I would like to thank all my colleagues for creating a supportive, encouraging, and fun environment to work. This journey would not have been possible without so many people. Special thanks to my friends who made this journey special, Ronak Shah, David Czudnochowski, Hannah Rosa Nesswetter, Rishi Saharia, Lotte Heckmann, Solange Gourdin, Jessica Girard, Aniek Siezenga, and many others.

Zusammenfassung

Deep-Learning-Modelle benötigen erhebliche Ressourcen für ihren Einsatz, was ihre breite Akzeptanz einschränkt. Das Ziel dieser Arbeit ist es, dieses Problem anzugehen, indem Methoden vorgeschlagen werden, um Deep-Learning-Modelle für Training und Einsatz effizienter zu machen.

Ein wichtiger Aspekt des maschinellen Lernens ist die Fähigkeit, visuelle Informationen aus begrenzten beschrifteten Daten zu verstehen, da groß angelegte Annotationsprozesse sehr teuer oder nicht durchführbar sein können. Im ersten Teil der Arbeit werden Methoden zur Verbesserung der Beschriftungseffizienz für Deep-Learning-basierte Computer-Vision-Aufgaben vorgeschlagen, wobei der Schwerpunkt auf zwei Ansätzen liegt - halbüberwachtes Lernen und aktives Lernen. Für das halbüberwachte Lernen schlägt die Arbeit einen Ansatz für die halbüberwachte semantische Segmentierung vor, der aus begrenzten, pixelweise annotierten Beispielen lernt und gleichzeitig zusätzliche annotationsfreie Bilder nutzt. Der vorgeschlagene Dual-Branch-Ansatz reduziert sowohl die Low-Level- als auch die High-Level-Artefakte, die typischerweise beim Training mit wenigen Labels auftreten, und seine Effektivität wird anhand mehrerer Standard-Benchmarks demonstriert. Für aktives Lernen wird in dieser Arbeit betont, dass die konventionellen Bewertungsschemata, die beim tiefen aktiven Lernen verwendet werden, entweder unvollständig oder unzureichend sind. Die Arbeit untersucht mehrere bestehende Methoden über viele Dimensionen hinweg und stellt fest, dass die untersuchten neuen zugrundeliegenden Faktoren für die Auswahl des besten aktiven Lernansatzes entscheidend sind. Die Arbeit bietet auch einen umfassenden Leitfaden für die Anwendung, um die beste Leistung für jeden Fall zu erzielen. Diese Arbeit befasst sich mit aktiven Lernmethoden für Bildklassifizierungs- und semantische Segmentierungsaufgaben.

Ein weiteres Problem bei tiefen neuronalen Netzen ist das katastrophale Vergessen, wenn sie sequentiell mit neuen oder sich entwickelnden Aufgaben konfrontiert werden. Dies macht sie für viele reale Anwendungen ungeeignet, da das Modellwissen mit allen jemals aufgetretenen Daten oder Aufgaben neu trainiert werden muss. Der zweite Teil der Arbeit konzentriert sich auf das Verständnis und die Behebung des katastrophalen Vergessens beim kontinuierlichen Lernen, insbesondere beim Class-incremental Lernen (CIL). Die Auswertung zeigt, dass eine Kombination einfacher Komponenten das katastrophale Vergessen

bereits in gleichem Maße beheben kann wie komplexere Maßnahmen, die in der Literatur vorgeschlagen werden.

Insgesamt bietet diese Arbeit rationalisierte Ansätze zur Verbesserung der Effizienz von Deep-Learning-Systemen und zeigt die Bedeutung vieler unerforschter Richtungen für eine verbesserte realistische Bewertung auf.

Abstract

Deep learning models require significant resources to deploy, limiting their widespread adoption. The aim of this thesis is to address this issue by proposing methods to make deep learning models more efficient for training and deployment.

One important aspect of machine learning is the ability to understand visual information from limited labeled data because large-scale annotation processes can be very expensive or infeasible. The first part of the thesis proposes methods to improve label efficiency for deep learning-based computer vision tasks focusing on two approaches - semi-supervised learning and active learning. For semi-supervised learning, the thesis proposes an approach for semi-supervised semantic segmentation that learns from limited pixel-wise annotated samples while exploiting additional annotation-free images. The proposed dual-branch approach reduces both the low-level and high-level artifacts typically encountered when training with few labels, and its effectiveness is demonstrated on several standard benchmarks. For active learning, the thesis emphasizes that conventional evaluation schemes used in deep active learning are either incomplete or below par. The thesis investigates several existing methods across many dimensions and finds that the studied new underlying factors are decisive in selecting the best active learning approach. The thesis also provides a comprehensive usage guide to obtain the best performance for each case. This thesis covers active learning methods for image classification and semantic segmentation tasks.

Another issue with deep neural networks is catastrophic forgetting when encountering new or evolving tasks in a sequential manner. The model must be retrained with all the data or tasks encountered to avoid forgetting, thus making them unsuitable for many real-world applications. The second part of the thesis focuses on understanding and resolving catastrophic forgetting in continual learning, particularly in the Class-incremental Learning (CIL) setting. The evaluation shows that a combination of simple components can already resolve catastrophic forgetting to the same extent as more complex measures proposed in the literature.

Overall, this thesis provides streamlined approaches to improve the efficiency of deep learning systems and highlights the importance of many unexplored directions for improved realistic evaluation.

Table of contents

List of Publications

[1] Sudhanshu Mittal, Silvio Galesso, and Thomas Brox. Essentials for class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3513–3522, June 2021.

[2] Sudhanshu Mittal, Joshua Niemeijer, Jörg P. Schäfer, and Thomas Brox. Best practices in active learning for semantic segmentation. *In DAGM German Conference on Pattern Recognition*, 2023.

[3] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high-and low-level consistency. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(4):1369–1379, 2019.

[4] Sudhanshu Mittal, Maxim Tatarchenko, Özgün Çiçek, and Thomas Brox. Parting with illusions about deep active learning. *arXiv preprint arXiv:1912.05361*, 2019.

[5] Maria A. Bravo, Sudhanshu Mittal, and Thomas Brox. Localized vision-language matching for open-vocabulary object detection. In *German Conference on Pattern Recognition (GCPR)*, 2022.

[6] Maria A. Bravo, Sudhanshu Mittal, Simon Ging, and Thomas Brox. Open-vocabulary attribute detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023.

Work done in [1–4] forms the basis of this thesis. Other publications [5, 6] are related to the topic but are not considered part of this thesis.

Chapter 1

Introduction

Deep learning has revolutionized our everyday life, empowering a wide range of services and products such as face recognition, speech recognition, virtual assistants, and self-driving cars. The advances in deep learning have provided significant improvements in understanding visual information. This can be noticed in the remarkable improvements in computer vision tasks like image classification, object detection, and semantic segmentation. Many of the benchmarks that were established five years ago have already achieved near-perfect performance, demonstrating the tremendous progress that has been made in this field.

Despite these great advances, deep learning faces several challenges that hinder its efficiency and scalability. One of the most significant challenges is the requirement for large-scale annotations to train deep neural networks. Labeling data can be very expensive since it is often laborious and time-consuming, especially for dense-prediction tasks such as object detection and semantic segmentation. Depending on the task, this cost may vary from a few seconds to a few hours. For example, the task of semantic segmentation is particularly costly in this regard, as it requires pixel-level annotations. Annotating a single image from a driving dataset can take an average of 1.5 hours [24]. In some application areas like medical imaging and genomics, labeled data can be scarce and hard to obtain, sometimes even impossible due to limited specialists in the field. In other applications, such as self-driving cars, raw data can be collected in abundance, but annotating this large raw data can still be costly. This highlights the need to develop new methods for learning with limited supervision. There exist many ways to tackle this challenge, such as semi-supervised learning, weakly-supervised learning, and active learning.

For many open-world deep-learning applications, data arrives as a continuous stream over time, and it is always changing, with new classes appearing and disappearing temporarily. Another limitation of deep neural networks is the inability to learn from such a continuous data stream. They tend to forget about previous tasks when they encounter a shift in the

training data or task, including new classes or domain changes. The most effective way to address this issue is to re-train the neural network with all the data combined together, which can be computationally expensive and memory-intensive. Therefore, there is a need to develop new techniques for continual learning, which can learn continuously without forgetting and without requiring re-training with all the data.

This thesis focuses on two major aspects of improving the learning efficiency of deep learning models for computer vision applications: label efficiency and training efficiency. First, the thesis investigates how to optimize the selection of images for annotation using active learning and proposes how to optimally use limited annotated samples using semi-supervised learning to improve label efficiency. Second, the thesis aims to identify the missing causes of catastrophic forgetting in a continual learning setting and propose solutions to address the limitations while minimizing memory requirements and training resources to improve training efficiency.

1.1 Learning with Limited Supervision

In deep learning, maximizing performance with minimal supervision is a crucial objective. There exist many learning techniques to fulfill this objective, including self-supervised learning, semi-supervised learning, weakly-supervised learning, and active learning. In this thesis, we focus on two techniques: semi-supervised learning and active learning. Both semi-supervised and active learning operate in a similar setting, where a large set of unlabeled data is available, and the aim is to maximize learning from a limited number of labeled samples. However, there is a subtle difference between the two objectives. Semi-supervised learning works on a post-hoc basis, where a small set of labeled samples is already given. In contrast, active learning is an ad-hoc approach, where the model must actively acquire this labeled data to maximize the model's performance.

1.1.1 Semi-supervised Learning:

In semi-supervised learning, the objective is to achieve maximum performance using a limited set of labeled samples in combination with a large set of unlabeled samples. Compared to supervised learning, semi-supervised learning has several clear advantages. One key benefit is that it can leverage both labeled and unlabeled data for learning, whereas supervised learning only uses labeled data for training. Relying on labeled data only can also be problematic if the labels provided by human annotators are incorrect or inconsistent. Moreover, semi-supervised learning is more robust to noisy data. This is because it leverages the underlying

structure of the data to provide more reliable signals than supervised learning. Typically, semi-supervised methods use a supervised learning objective to learn from labeled data and an unsupervised learning objective to learn from unlabeled data. This enables them to learn from a vast amount of data without incurring a large annotation cost.

While image classification has been extensively studied in a semi-supervised learning setting for many years, dense pixel-level classification with limited supervision has only recently drawn attention. Learning with limited supervision for dense pixel-level tasks like semantic segmentation is also a more significant challenge compared to image classification, as the cost of annotation is much higher. In particular, semi-supervised learning for dense tasks like semantic segmentation demands a more detailed understanding of visual content and the ability to distinguish between objects that may be partially obscured or share similar features. Given these challenges, this thesis focuses on semi-supervised learning specifically for the semantic segmentation task.

There are two main types of semi-supervised learning techniques [130, 131, 110] for image classification tasks. The first one involves using predictions as pseudo-labels for unlabeled data and ground-truth labels for labeled data. This type of approach is referred to as the self-training approach. The second approach is to learn from unlabeled samples using consistency loss across two different views of an image. Both approaches use strong augmentations, including affine transformations and photometric augmentations, using a student-teacher approach. However, these strong augmentation operations are not directly applicable to semi-supervised segmentation methods due to the absence of low-density regions along class boundaries [139]. Consequently, it would violate the smoothness or clustering assumption of semi-supervised learning. Additionally, pure pseudo-labeling-based solutions suffer from the problem of confirmation bias since only the highly confident predictions are used as pseudo-labels, which can lead to the bootstrapping of incorrect pseudo-labels and suppression of low-frequency classes. This problem is more severe for class-imbalanced datasets. Therefore, novel techniques are needed to address the unique challenges of semi-supervised learning for semantic segmentation tasks.

Our proposed method is motivated by the observation that models trained with only a small labeled set of images tend to produce inaccurate predictions for both low-level and high-level details in the image. At the low level, the model often struggles to capture the object shapes and produces incoherent surfaces with holes, as well as inaccurate boundaries. At the high level, it tends to assign large regions of object instances to incorrect classes. Figure 2.1 shows an example of low-level and high-level artifacts observed for a model trained with only a few labeled samples. In this thesis, we propose a two-branch model to

address both low-level and high-level artifacts separately. Moreover, our method does not rely on strong augmentation techniques.

To address low-level artifacts, we propose a generative adversarial network (GAN) mode where the segmentation network acts as a generator, and the discriminator identifies predicted segmentation maps from ground-truth segmentation maps. This approach helps in learning low-level features similar to ground-truth data. We additionally propose an auxiliary self-training approach based on the whole predicted segmentation map on unlabeled data using the discriminator scores. This self-training approach promotes faster learning of the generator model and it also automatically allows low-confidence pixel-level predictions for creating pseudo-labels because the whole predicted segmentation mask with an overall high rating is selected for self-training training. To address high-level artifacts, we utilize the well-designed multi-label classification model to predict the classes present in an image. The class-level prediction, which is more robust than the segmentation model itself, helps to rectify false positive predictions by masking out low-scoring classes from the segmentation prediction (see Figure 2.1).

The proposed method is evaluated on three standard semantic segmentation benchmarks with varying amounts of labeled data. To the best of our knowledge, this is the first work to propose an end-to-end trainable model for semi-supervised semantic segmentation. Our method not only outperforms all previous methods but also demonstrates competitive performance in comparison to the latest methods proposed after our work. The model and training details, along with ablation studies, are presented in Chapter 2. In the semi-supervised learning setting, a limited set of labeled samples is typically provided. However, it is possible to make better use of the annotated budget by smartly selecting which samples to label, assuming that some samples are more valuable for the model's performance. The idea of selecting samples for annotation is studied under a topic called active learning.

1.1.2 Active Learning

Active learning is a technique that can significantly reduce the labeling cost of training machine learning models by selecting the most valuable samples for annotation while maximizing the performance on the given task. The active learning cycle for deep networks starts from a large pool of unlabeled data samples. Unlike traditional non-deep learning based active learning methods that select a single sample for annotation, a pool of samples is chosen for annotation due to practical considerations in deep learning. The selection of the pool of samples is based on a user-defined acquisition function that attempts to measure the value of the samples, and the model is trained on the annotated samples. The acquisition function is used again to select more samples, and the active learning cycle continues. In practice, the

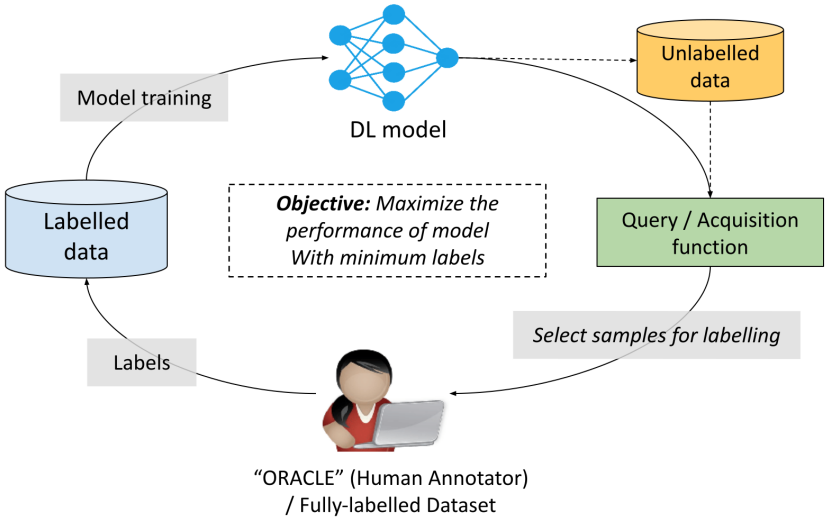


Fig. 1.1 Active Learning cycle.

selection is often done in conjunction with model training for maximum efficiency *i.e.*, the acquisition function can be defined using the last trained model in the previous cycle. The active learning cycle continues until an acceptable performance is achieved or the annotation budget is exhausted (See Figure 1.1).

Despite over a decade of research work, deep active learning remains an open challenge, and there still exists an ambiguity concerning selecting the most appropriate active learning method for a given dataset. This is partially due to the difficult nature of the problem and inadequate evaluation schemes. Measuring the value of sample annotation for a model is a non-trivial task that depends on a combination of factors, including the objective of the task, the current state of the model, data distribution, annotation budget as well as regularization and optimization techniques.

Variability in the data distribution plays a major role in active learning method selection. If the annotation budget is too small, bias towards this limited set of data may result in poor selection by the active learning method, leading to decreased model performance. Whereas if the dataset size is very large with a lot of redundancies, the active learning method may select similar samples due to the limited prior knowledge about the dataset, which again leads to reduced performance. Active learning also requires balancing the trade-off between exploration and exploitation of the unlabeled data to avoid overfitting and underfitting. In early training stages, the model may struggle to learn if presented with too difficult samples, leading to slower learning, while overly simplistic samples later on may also result in slower learning. Therefore, selecting an optimal acquisition function is highly dependent on prior knowledge about the dataset and the current optimization state. This way, deep active learning

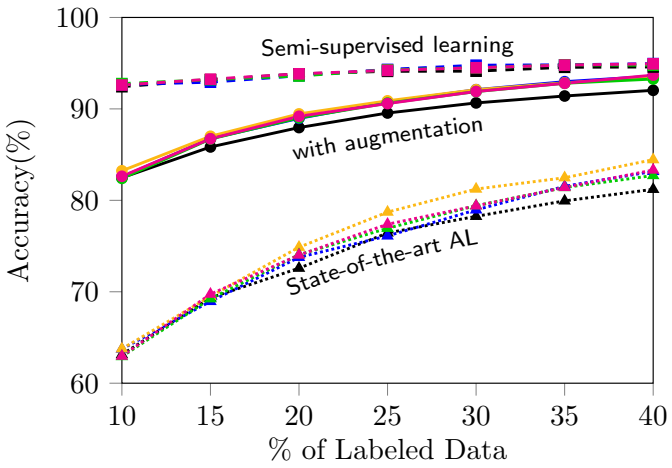


Fig. 1.2 State-of-the-art active learning methods do not consistently use modern data augmentation techniques or advances in the closely related field of semi-supervised learning, which leads to the wrong impression about the current state of the field. Results are shown for image classification on CIFAR 10.

is also connected to curriculum learning with additional control over data selection. In this thesis, we analyze active learning methods across several factors, including data distributions in terms of levels of redundancy, range of annotation budgets, and learning objectives across multiple datasets and tasks.

In traditional deep active learning, a large pool of unlabeled samples is used to select a batch of the most valuable samples for annotation. However, this large pool of unlabeled samples can be easily utilized for semi-supervised learning along with the already annotated samples. This utilization of unlabeled samples for learning has been largely ignored in active learning research. To address this gap, in this thesis, we propose to integrate the latest semi-supervised learning methods into active learning and conduct a detailed study of this combination.

Most of the current research on deep active learning has mainly focused on image classification tasks. However, the challenge of high annotation cost becomes critically more important for dense prediction tasks like semantic segmentation, as noted earlier regarding the semi-supervised learning problem. In this thesis, we study deep active learning methods for both image classification and semantic segmentation tasks. We seek answers to specific missing questions that have not been explored by previous studies, such as the effectiveness of AL methods w.r.t. diverse data distributions, the impact of data regularization, the integration of semi-supervised learning, and the influence of various annotation budget settings on model performance.

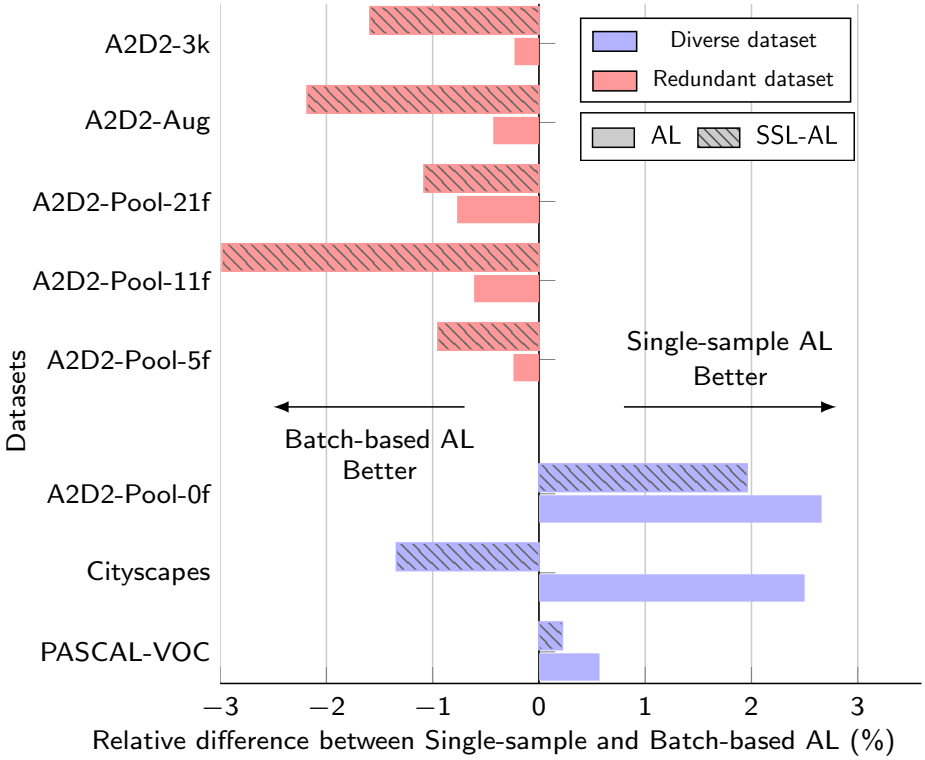


Fig. 1.3 The figure summarizes and compares several AL methods on datasets with different levels of redundancy. The figure shows that the best-performing AL method changes depending on the AL objective in different conditions and with the integration of semi-supervised learning with active learning. The figure shows the difference between the best-performing single-sample-based AL method and best performing batch-based AL method, which is discussed in detail in Chapter 3.3.6.

We show deep active learning methods for image classification improve by a large margin when integrated with data augmentation and semi-supervised learning. However, they are highly inconsistent across different settings, and they only marginally perform better than the random selection baseline (see Figure 1.2). These findings have important implications for the successful development of active learning methods and highlight the need for better evaluation workflows. As a result of the study, we propose an evaluation protocol for deep active learning methods. Chapter 3.2 covers the details of this study of active learning methods for image classification tasks.

We found that the existing benchmarks and methods for active learning for semantic segmentation only cover a limited range of realistic scenarios, leading to incomplete and sometimes misleading conclusions. We show that conclusions drawn from these previous works may not be generalizable to different realistic settings. Our study shows that data distribution is decisive for the performance of various active learning objectives proposed in the literature. Particularly, redundancy in the data, as it appears in most driving scenarios

and video datasets, plays a large role. We also demonstrate that the integration of semi-supervised learning with active learning can improve performance, although this integration is more complex than in the case of the image classification task. Figure 1.3 summarizes the performance of the best active learning method across several datasets with different levels of redundancy. As an outcome of our extensive study, we provide a comprehensive usage guide to obtain the best performance for diverse settings and also propose an exemplary evaluation task to showcase the practical implications of our research findings. Chapter 3.3 contains a detailed study of active learning methods for the semantic segmentation task.

1.2 Improving Training Efficiency

Modern neural networks are limited by their ability to learn from evolving streams of training data. When neural networks are trained sequentially on new or evolving tasks, their accuracy drops sharply on previously learned tasks, making them unsuitable for many real-world applications. This phenomenon, referred to as catastrophic forgetting, is attributed to the change in model parameters while solving a new task.

A brute-force way to deal with this challenge would be to collect and annotate all the data, then train the model repeatedly. However, this approach is certainly not practical due to reasons like memory restrictions, data security restrictions, computational expenses, and sustainability issues. Therefore, the goal of continual learning is to adapt the model continually to new tasks while accumulating knowledge from them without disrupting previous knowledge. This should be achieved without requiring the model to be re-trained every time it encounters new data or tasks. More precisely, we begin with a model, which is trained for a particular task, and the objective is to learn a new task without losing the ability to perform the previous task.

This thesis focuses mainly on a continual learning scenario for image classification known as class-incremental learning, where the objective is to learn a completely new set of classes without access to the data of the old set of classes while still retaining the ability to do inference on all the classes seen until then. Some methods optionally consider a small amount of stored data from previous tasks.

Although the phenomenon of catastrophic forgetting is well-known, but its underlying reasons have not yet been fully understood. One of the main causes of catastrophic forgetting in class-incremental learners includes weight drift in the last layers, where the network's weights are updated to learn the new task, thus causing an imbalance between the weights responsible for old and new classes. Therefore, we propose a compositional model to address

the issue of bias in weight vectors. Our model isolates the underlying issue and combines the simple and effective components to build a robust model.

The problem of class-incremental learning is closely related to transfer learning but with the additional constraint of remembering and merging the old classes with new ones. Given a pre-trained model, the model parameters are less likely to be overwritten by the new task if they are easily transferable to the new task. However, there exists no exact metric to measure how transferable the model's parameters are to a new task. Our study finds that overfitting of the model parameters on the new task is not the only reason for forgetting but also overfitting on the initial task. We show that the more overfitted models are likely to forget more in the incremental steps. Therefore, the quality of learned representation in terms of transferability plays a major role in avoiding forgetting. We propose a proxy metric for measuring the transferability of the model parameters for the studied problem.

In class-incremental learning, the granularity of the final task becomes finer as more classes are observed. Therefore, preserving inter-class information in the model becomes highly relevant. One way to measure this inter-class information within a sample is through the correctness of the non-maximum output logits for the sample. This information is also known as dark knowledge in the literature and is referred to as secondary-class information. Our study shows that the secondary-class information is a good indicator of the transferability of the model's parameters for class-incremental learning. Chapter 4 provides details for the proposed compositional model and shows how the quality of learned representation is connected to catastrophic forgetting in class-incremental learning.

Chapter 2

Semi-supervised Semantic Segmentation

The content of this chapter was adapted from the following paper.

> Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high-and low-level consistency. IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(4):1369–1379, 2019.

All co-authors mainly contributed to the project discussions as well as the final paper text editing. All the other contributions described in this chapter are made by myself.

* * *

Motivation. Deep Neural Networks, including CNNs [72] and Transformers [129] have demonstrated excellent results on the semantic segmentation task for several different datasets [19, 20, 97, 141, 146]. However, this success usually comes at the cost of collecting dense pixel-wise annotations - a cumbersome process that involves much manual effort. Attempting to alleviate this limitation, several methods exploit various weaker forms of supervision, including image-level labels [8, 98, 124], bounding boxes [27, 94], scribbles [75, 114], or, recently, image-text pairs [133]. Since acquiring unlabeled data is cheaper, for *e. g.* from the web, recently several works [57, 111, 22, 54, 123] have also considered semi-supervised learning for semantic segmentation. In this semi-supervised learning setting, the objective is to learn from a limited set of fully-annotated images and a large set of completely annotation-free images. In this chapter, we propose a semi-supervised learning method for semantic segmentation.

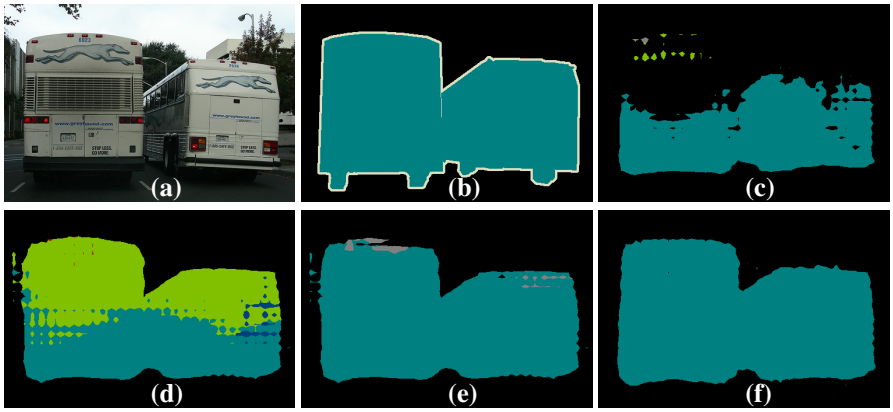


Fig. 2.1 An image from the PASCAL VOC dataset (a) and its ground-truth segmentation mask (b). Prediction (c) is obtained with supervised training on 5% labeled samples. Using the other 95% unlabeled images, our GAN-based branch improves the shape estimation (d). The second branch adds high-level consistency by removing false positives (e). (f) shows the output when training on 100% pixel-wise labeled samples.

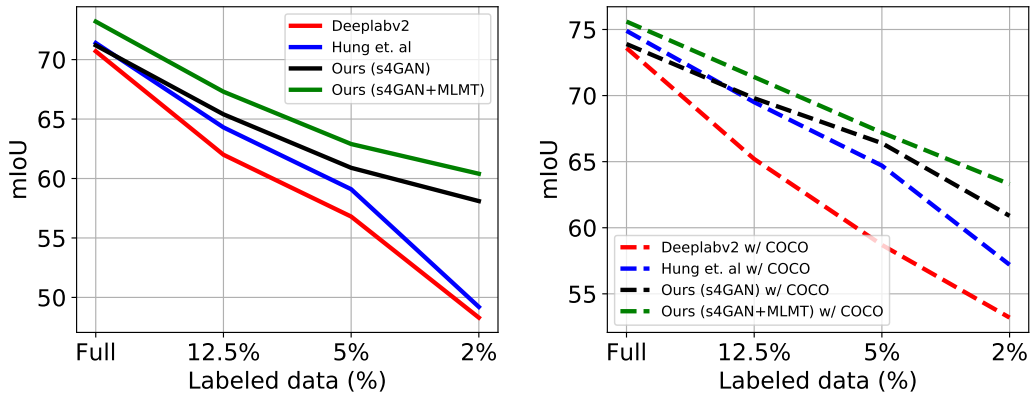
Problem definition. Assuming we have a dataset consisting of n examples, denoted as $\mathcal{D} = (\mathbf{x}, \mathbf{y}$, where $\mathbf{x}$ is the set of input images, and $\mathbf{y}$ is the set of corresponding pixel-wise segmentation label. Let $\mathcal{D}^\ell = \{\mathbf{x}^\ell, \mathbf{y}^\ell\}$ denote the set of labeled examples, and $\mathcal{D}^u = \{\mathbf{x}^u\}$ denote the set of unlabeled examples. Thus, $\mathcal{D}^\ell$ and $\mathcal{D}^u$ partition the dataset $\mathcal{D}$, i.e., $\mathcal{D} = \mathcal{D}^\ell \cup \mathcal{D}^u$ and $\mathcal{D}^\ell \cap \mathcal{D}^u = \emptyset$. The objective of semi-supervised learning is to learn a function $f : \mathcal{X} \to \mathcal{Y}$ that maps input images $\mathbf{x} \in \mathcal{X}$ to labels $\mathbf{y} \in \mathcal{Y}$, where $\mathcal{X}$ is the space of images, and $\mathcal{Y}$ is the space of labels. The function f is learned by minimizing a loss function $\mathcal{L}(f; \mathcal{D}^\ell, \mathcal{D}^u)$ that measures the discrepancy between the predictions of f and the true labels in $\mathcal{D}^\ell$, as well as the predictions of f on the unlabeled examples in $\mathcal{D}^u$. Formally, the semi-supervised learning problem can be written as the following optimization problem:

$$\min_f \mathcal{L}(f; \mathcal{D}^\ell, \mathcal{D}^u) \tag{2.1}$$

where the loss function $\mathcal{L}$ is defined as the sum of two sets of terms: a set of terms for supervised learning $\mathcal{L}_{sup}(f; \mathcal{D}^\ell)$ and a set of terms for unsupervised learning $\mathcal{L}_{unsup}(f; \mathcal{D}^u)$:

$$\mathcal{L}(f; \mathcal{D}^\ell, \mathcal{D}^u) = \mathcal{L}_{sup}(f; \mathcal{D}^\ell) + \mathcal{L}_{unsup}(f; \mathcal{D}^u) \tag{2.2}$$

Modes of failure. CNNs trained on limited data are subject to two typical modes of failure; see Figure 2.1(c-d). The first one appears as inaccuracy in low-level details, such as wrong object shapes, inaccurate boundaries, and incoherent surfaces. The second one is the



(a) (b)

Fig. 2.2 **Semi-supervised Semantic Segmentation**: The proposed semi-supervised learning (SSL) approach improves over the baselines when only little labeled data is available using unlabeled data, especially with less than 5% labeled samples. Performance is shown on the PASCAL VOC dataset without (a) and with (b) COCO pre-training.

misinterpretation of high-level information, which leads to assigning large image regions to the wrong classes.

Our approach. Our two network branches are designed to address these two types of artifacts separately. To deal with low-level errors, we propose an improved GAN-based model, where the segmentation network acts as a generator. It is trained together with a discriminator that discriminates between generated and ground-truth segmentation maps. Instead of using the original GAN loss, which causes instability, we propose to use the feature-matching loss introduced by Salimans *et al.* [102]. Moreover, we introduce the self-training procedure based on the discriminator score, which stabilizes the adversarial learning process and improves the performance of the resulting model.

To resolve the second type of artifact, we propose a semi-supervised multi-label classification branch that decides which classes are present and which ones are missing in the image and thus aids the segmentation network in making globally consistent decisions with the first branch. To utilize extra image-level information from unlabeled images, we leverage the success of ensemble-based semi-supervised classification (SSL) methods [70, 116]. The two branches act in a complementary manner and successfully fix both low-level and high-level errors; see Figure 2.1 for a typical example.

Scope of this chapter. We demonstrate the effectiveness of our approach on different amounts of labeled data across a range of popular semantic segmentation datasets: PASCAL VOC 2012 [34], PASCAL-Context [89] and Cityscapes [24]. We consistently achieve the best

results compared to existing methods and define the new state of the art in semi-supervised semantic segmentation. Our approach proves particularly efficient when only very few training samples are available: with as little as 2% labeled data on the PASCAL-VOC dataset, we report an 11% performance improvement over state-of-the-art (see Figure 2.2). Chapter 2.4 covers the experimental results on the mentioned three benchmarks with detailed ablation studies. Our approach is one of the first to provide an end-to-end learning-type solution to the semi-supervised semantic segmentation task. We also compare our work to some of the latest state-of-the-art methods [54, 22, 123, 132] and find that it still performs competitively to these new methods when trained with similarly high-resolution input images and new data augmentation techniques. Our approach offers a few additional advantages. Our self-training technique uses complete prediction as the pseudo-label for unlabeled samples. Therefore, the method is less prone to training bias [17] caused due to self-training. Our approach is also flexible regarding the type of supervision provided to the model. Chapter 2.4.2 shows that the approach can easily utilize extra weak supervision in the form of image labels and scribbles. And the model with weak supervision compares favorably to the existing methods operating in the same setting. The source code of this chapter is available [1].

2.1 Related work

Weakly-supervised and Semi-supervised Segmentation. To reduce annotation effort, many existing approaches rely on weakly- and semi-supervised training schemes which use weak labels from the whole dataset like image-level class labels [94, 125], bounding boxes [27, 64, 94] or scribbles [75, 114], where semi-supervised schemes [64, 94, 125] additionally use a few pixel-wise segmentation labels. Only two works [57, 111], prior to this work, considered true semi-supervised learning, i.e., they improve semantic segmentation with completely annotation-free images. These methods, like ours, utilize a GAN-based model. Although both approaches use the GAN in a different manner. Souly *et al.* [111] uses the GAN to generate additional images to enhance the features learned by the segmentation network and further extend their semi-supervised method by generating additional class-conditional images. Most related to ours is the work by Hung *et al.* [57]. They also propose a GAN-based design that enables learning from unlabeled samples. Also, Luc *et al.* [80] share some common ground with our work, although their work does not comprise semi-supervised learning. In their case, the GAN replaces CRF-post-processing, which enhances low-level consistency in the segmentation maps.

[1] Source code: https://github.com/sud0301/semisup-semseg

Recent follow-up works. Semi-supervised semantic segmentation has gained a lot of interest. Many interesting follow-up works have been proposed lately. Self-training and strong augmentation techniques such as CutMix [139] have become the main components of these new methods. The latest methods utilize self-training with a student-teacher setup, where predictions on unlabeled samples are partially selected, based on a confidence threshold, and used as pseudo labels for training. Recently Wang *et al.* [123] proposed (U2PL) to make use of low-confidence predictions as well, instead of completely filtering them out, using contrastive learning. AEL [54] approach argues that self-training harms underperforming categories due to data imbalance and focuses on long-tailed label distribution. AEL adaptively balances the training of well and poor-performing categories. This technique proves to be useful, especially for class-imbalanced datasets like Cityscapes. Xu *et al.* [132] proposes an approach with an additional prototype-based predictor to learn within-class feature distributions. This is, so far, the best-performing model in the literature.

Semi-supervised Classification. In contrast to segmentation, many semi-supervised methods exist for image classification [9, 70, 87, 102, 116]. Oliver *et al.* [92], however, criticizes that most of the work lacks realistic evaluation to address real-world conditions. In order to rectify the issue, they propose a new experimental methodology closer to real-world settings. We find that new consistency-based semi-supervised classification methods [9, 116] show improvement over the supervised baseline while satisfying at least two procedures mentioned by Oliver *et al.* [92]. Firstly, those methods show improvement over the supervised setting while using a high-quality supervised baseline. Secondly, they can improve upon the pre-trained network using unlabeled data. In this work, we utilize a consistency-based approach, which uses a mean-teacher [116] model, for the second branch.

Network Fusion. The approach to fuse spatial and class information by channel-wise selection is inspired by some recent works in other domains. Hu *et al.* [55] proposed SE-Net for image classification, which learns to combine spatial and channel-wise information by calibrating channel-wise feature maps. Following SE-Net, Zhang *et al.* [141] proposed to incorporate class information in semantic segmentation to highlight class-dependent feature maps. Multiple works [51, 124, 125] have explored the usage of classification methods, both in a shared and a decoupled manner, to constructively use class information for semi- and weakly supervised semantic segmentation. Our work uses a decoupled approach with a late fusion of spatial and class information to remove false positive class channels.

2.2 Our Method

We propose a two-branch approach to the task of semi-supervised semantic segmentation as shown in Figure 2.3. The upper branch predicts pixel-wise class labels and is referred to as the *Semi-Supervised Semantic Segmentation GAN* (s4GAN). The lower branch performs image-level classification and is denoted as the *Multi-Label Mean Teacher* (MLMT).

The core of the s4GAN branch is a standard segmentation network for generating per-pixel class labels given the input image. We combine conventional supervised training with adversarial training, which allows leveraging unlabeled data to improve the generalization of the model. The segmentation network acts as a generator and is trained together with a discriminator responsible for distinguishing the ground truth segmentation maps from the generated ones. We additionally treat the output of the discriminator as a quality measure and use it to identify good quality predictions, which are further exploited for self-training.

The MLMT branch predicts image-level class labels used to filter the s4GAN outputs. Its core is a mean-teacher classification model, which is an online ensemble of the student classification model. MLMT branch is trained in a semi-supervised manner using standard classification loss and consistency loss. The MLMT effectively removes false positive predictions of the segmentation network. The contributions of the two branches MLMT and s4GAN are complementary to each other. The outputs of the two branches are combined to produce the final result.

2.2.1 s4GAN for Semantic Segmentation

In our s4GAN model, the segmentation network S acts as a generator network that takes image $\mathbf{x}$ as input and predicts C segmentation maps, one for each class. The discriminator D gets the concatenated input of the original image and its corresponding predicted segmentation. Its task is to match the distribution statistics of the predicted and the real segmentation maps.

Training S The segmentation network S is trained with a loss $\mathcal{L}_S$, which is a combination of three losses: the standard cross-entropy loss, the feature matching loss, and the self-training loss.

Cross-entropy loss. This is a standard supervised pixel-wise cross-entropy loss term $\mathcal{L}_{ce}$. The loss for the output $S(\mathbf{x})$ of size $H \times W \times C$ is evaluated only for the labeled samples $\mathbf{x}^\ell$:

$$\mathcal{L}_{ce} = -\sum_{h,w,c} \mathbf{y}^\ell(h,w,c) \log S(\mathbf{x}^\ell)(h,w,c), \tag{2.3}$$

where $\mathbf{y}^\ell$ is the ground-truth segmentation mask.

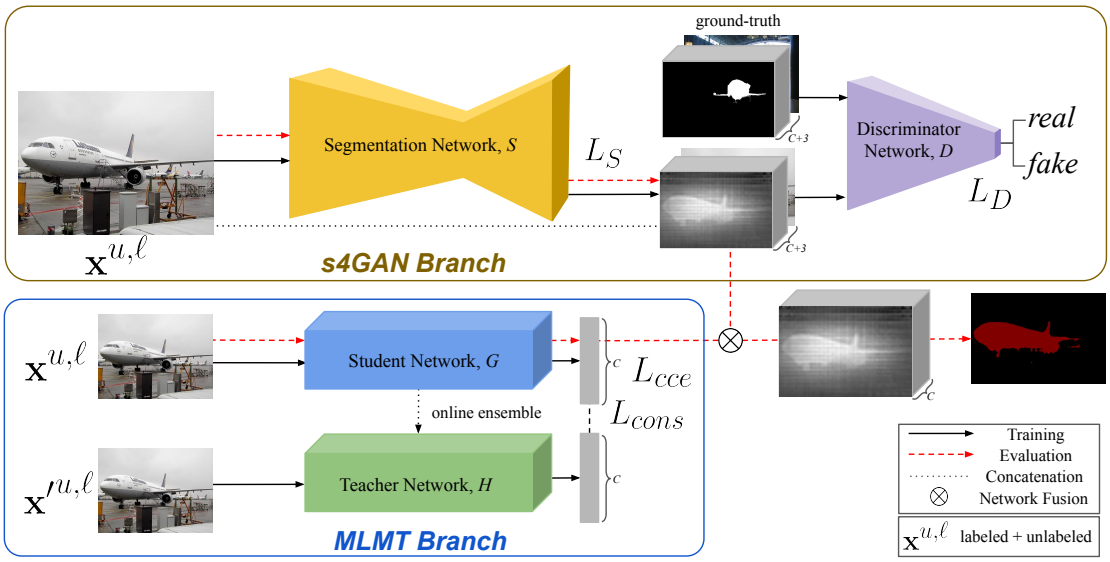


Fig. 2.3 Overview of our proposed semi-supervised segmentation approach. The s4GAN branch is a GAN-based model which improves the low-level details in the segmentation prediction. The MLMT branch performs semi-supervised multi-label classification to exploit class-level information for removing false-positive predictions from the segmentation map.

Feature matching loss. The feature matching loss $\mathcal{L}_{fm}$ [102] aims to minimize the mean discrepancy between the feature statistics of the predicted, $S(\mathbf{x}^u)$ and the ground-truth segmentation maps, $\mathbf{y}^\ell$:

$$\begin{aligned}\mathcal{L}_{fm} = &\| \mathbb{E}_{(\mathbf{x}^\ell,\mathbf{y}^\ell)\sim\mathcal{D}^\ell}[D_k(\mathbf{y}^\ell \oplus \mathbf{x}^\ell)] \\ &- \mathbb{E}_{\mathbf{x}^u\sim\mathcal{D}^u}[D_k(S(\mathbf{x}^u) \oplus \mathbf{x}^u)] \|,\end{aligned} \tag{2.4}$$

where $D_k(\cdot)$ is the intermediate representation of the discriminator network after the k^{th} layer. Both ground-truth and predicted segmentation masks are concatenated with their corresponding input images. Intuitively, it encourages the generator to predict segmentation maps that have the same feature statistics as the ground truth and, therefore, also qualitatively resemble the ground truth. This loss is used on the unlabeled samples $\mathbf{x}^u$, thus forcing plausible solutions even for cases where dense labels are unavailable. This loss is class-agnostic, unlike the self-training loss, and encourages the segmentation model to produce accurate low-level predictions like edges and corners.

Self-training loss. During GAN training, the discriminator (D) and the generator (G) networks need to be balanced. If D starts off being too strong, it does not provide any useful learning signal for G. In order to facilitate such balanced dynamics, we introduce the self-training (ST) loss. The main idea is to pick the best generator outputs (i.e., those able

to fool D) which do not have the corresponding ground truth and reuse them for supervised training. Intuitively, this pushes G more to produce predictions that D cannot distinguish from the real ones. This impedes the progress of D and does not allow it to become too strong quickly. This self-training loss re-establishes the desired balance between the generator and the discriminator model.

Technically, the output of D varies between 0 and 1, where 0 should be assigned to the predicted segmentation maps and 1 to the ground-truth segmentation maps. We use this score as a confidence measure for the quality of the predicted segmentation. High-quality predictions are used for supervised training by creating pseudo-labels, i.e., we calculate the standard cross-entropy loss based on them. The self-training loss term $\mathcal{L}_{st}$ is thus defined as:

$$\mathcal{L}_{st} = \begin{cases} -\sum\limits_{h,w,c} \mathbf{y}^* \log S(\mathbf{x}^u), & \text{if } D(S(\mathbf{x}^u) \geq \gamma \\ 0, & \text{otherwise,} \end{cases} \tag{2.5}$$

where γ is the confidence threshold which controls how certain D needs to be about the prediction in order for it to be used in self-training; $\mathbf{y}^*$ are the pseudo-pixel-wise labels generated from the prediction $S(\mathbf{x}^u)$ of the segmentation network.

The final training objective L_S is composed of the three described terms:

$$\mathcal{L}_S = \mathcal{L}_{ce} + \lambda_{fm}\mathcal{L}_{fm} + \lambda_{st}\mathcal{L}_{st}, \tag{2.6}$$

where $\lambda_{fm}, \lambda_{st} > 0$ are the corresponding weights.

Training D The objective of the discriminator is to distinguish between the ground-truth (real) and predicted (fake) segmentation masks. If the quality of the predicted mask is good, then the discriminator is likely to fail in its task, whereas poor prediction quality would result in a reduction of model D's loss. This encourages the segmentation network to produce better predictions using the feature-matching loss as described above. The discriminator network is trained with the original GAN objective as proposed by Goodfellow *et al.* [46]

$$\begin{aligned} \mathcal{L}_D =& \mathbb{E}_{(\mathbf{x}^\ell, \mathbf{y}^\ell) \sim \mathcal{D}^\ell}[\log D(\mathbf{y}^\ell \oplus \mathbf{x}^\ell)] \\ &+ \mathbb{E}_{\mathbf{x}^u \sim \mathcal{D}^u}[\log(1 - D(S(\mathbf{x}^u) \oplus \mathbf{x}^u))], \end{aligned} \tag{2.7}$$

where $\oplus$ denotes concatenation along the channel dimension. Following the original GAN idea, D learns to differentiate between the real $\mathbf{y}^\ell$ and the fake segmentation masks $S(\mathbf{x}^u)$ concatenated with the corresponding input images.

2.2.2 Multi-label Semi-supervised Classification

We extend an online ensemble-based semi-supervised classification method (Mean-Teacher) [116] for semi-supervised multi-label image classification. This model consists of two networks: a student network G and a teacher network H. Both networks receive the same images under different small perturbations. The weights (θ') of the teacher network are the exponential moving average (online ensemble) of the student network's weights (θ). The predictions made by the student model are encouraged to be consistent with the predictions of the teacher model using the consistency loss, which is the mean-squared error between the two predictions.

We optimize the student network using the categorical cross-entropy loss L_{cce} for labeled samples $\mathbf{x}^{\ell}$, and using the consistency loss L_{cons} for all available samples ($\mathbf{x}^{u,\ell}$):

$$\begin{aligned}\mathcal{L}_{MT} = & \underbrace{-\sum_{c} \mathbf{z}^{\ell}(c) \log(G_{\theta}(\mathbf{x}^{\ell})(c))}_{L_{cce}} \\ & + \lambda_{cons} \underbrace{\|G_{\theta}(\mathbf{x}^{(u,\ell)}) - H_{\theta'}(\mathbf{x'}^{(u,\ell)})\|^2}_{L_{cons}},\end{aligned} \tag{2.8}$$

where $\mathbf{x}$ and $\mathbf{x}'$ are differently augmented images for student and teacher networks, respectively, $\mathbf{z}^{\ell}$ is the multi-hot vector for ground-truth class labels. The parameter $\lambda_{cons} > 0$ controls the weight of the consistency loss in $\mathcal{L}_{MT}$.

2.2.3 Network Fusion

The two described branches are trained separately. For evaluation, the output of the classification branch simply deactivates the segmentation maps of those classes not present in the input image:

$$S(\mathbf{x})_c = \begin{cases} 0 & \text{if } G(\mathbf{x}_c) \leq \tau \\ S(\mathbf{x})_c & \text{otherwise} \end{cases} \tag{2.9}$$

where $S(\mathbf{x})_c$ is the segmentation map for class c, $G(\mathbf{x})_c$ is the soft output of the MLMT-branch, and $\tau = 0.2$ is a threshold on that softmax output obtained by cross-validation.

2.3 Experiment setup

2.3.1 Datasets

PASCAL VOC 2012. It is a standard object-centric semantic segmentation dataset. The dataset consists of 20 foreground object classes and one background class. We use the augmented annotation set, which consists of 10582 training images and 1449 validation images. The training set contains 1464 images from the original PASCAL data, and 9118 extra images from the Segmentation Boundary Dataset (SBD) [48]. The training data augmentations include random resizing, cropping to 321×321, and horizontal flipping. All the results for the PASCAL VOC dataset are shown on the validation set.

PASCAL-Context. This is a whole scene parsing dataset containing 4,998 training and 5,105 testing images with dense semantic labels. Following the previous work [19, 76, 141], we used semantic labels for the 60 most frequent classes, including the background class. The training data augmentations were the same as for the PASCAL VOC dataset.

Cityscapes. This is an urban driving scene dataset with 2975, 500, and 1525 densely annotated images for training, validation, and testing, and it contains 19 classes. We downsample the original 1024×2048 images by a factor of 2 to fit the models in the GPU memory. The training data is augmented with random crops of size 256×512 and horizontal flipping. All the results on the Cityscapes dataset are shown on the validation set.

Evaluation Metric. We report the mean Intersection-over-Union (mIoU) for all our experiments as the evaluation metric.

2.3.2 Network Architecture

Semi-supervised Segmentation GAN. We used DeepLabv2 [19] as our main segmentation network for the comparison with previous methods and ablation studies. Due to memory constraints, we used a single-scale variant of it. Later, we used DeepLabv3+ [21] to compare with the latest methods, which are based on the same architecture. The discriminator network of the GAN model was a standard binary classification network consisting of 4 convolutional layers with 4×4 kernels with $\{64, 128, 256, 512\}$ channels, each followed by a Leaky-ReLU [82] activation with negative slope of 0.2 and a dropout [112] layer with dropout probability of 0.5. We found this high dropout rate to be crucial for stable GAN training. The last convolutional layer is followed by global average pooling and a fully-connected layer. The output vector representation produced after global average pooling is used for evaluating the feature matching loss.

Semi-supervised Multi-label Classification Network. We used ResNet-101 [49] pre-trained on the ImageNet dataset [29] as the base architecture. We replaced the softmax activation layer with a sigmoid function for each class, for multi-label classification.

2.3.3 Training details

Similar to [19], we used the poly-learning policy for both the segmentation and the discriminator networks of the GAN model, where the base learning rate was multiplied by a factor of $((1-\frac{\text{iter}}{\text{max\_iter}})^{pow})$ in every iteration. In our setup, $pow = 0.9$. Following the learning scheme in [57], the segmentation network was optimized using the SGD optimizer with a base learning rate of 2.5e-4, momentum 0.9, and a weight decay of 5e-4. The discriminator network was optimized using the Adam optimizer [67] with a base learning rate of 1e-4 and betas set to $(0.9, 0.99)$. The model was trained for 35K iterations on the PASCAL VOC and Cityscapes dataset, and for 50K iterations on the PASCAL-Context dataset. All the learning hyper-parameters remained the same for all datasets except for the Cityscapes dataset, where the base learning rate of the discriminator network was set to 1e-5. We used a batch size of 8 for both PASCAL datasets and a batch size of 5 for the Cityscapes dataset. Through cross-validation, we find the optimal loss weights: $\lambda_{fm} = 0.1$, $\lambda_{st} = 1.0$, $\lambda_{cons} = 1.0$ and $\tau = 0.2$. These hyper-parameters remained the same for all datasets, whereas we set $\gamma = 0.6$ for both PASCAL datasets and 0.7 for the Cityscapes dataset. Overall, the gamma parameter is fairly robust: the performance varies within the range of 0.4% for gamma values between 0.5 and 0.8 on the Cityscapes dataset. Our implementation is based on the open-source toolbox Pytorch [95]. All the experiments were run on a Nvidia Tesla P100 GPU.

2.3.4 Baselines

We compare to the DeepLabv2 [19] network as the fully-supervised baseline approach, which was trained only on the labeled part of the dataset. DeepLabv2 makes use of dilated convolutions to enlarge the receptive field size and incorporate a larger context and introduces atrous spatial pyramidal pooling (ASPP) to capture image context at multiple levels.

In Table 2.1, we compare our methods to the semi-supervised baseline proposed by Hung *et al.* [57]. This was the only work proposed before our method contribution. Apart from the differences described in Section 2.1, they also use a two-stage GAN training. In the first stage, both D and G are trained only using labeled data. In the second stage, D's outputs are used to update G using unlabeled samples, while D itself is further trained only on the labeled images. All the methods are trained with the same data augmentation.

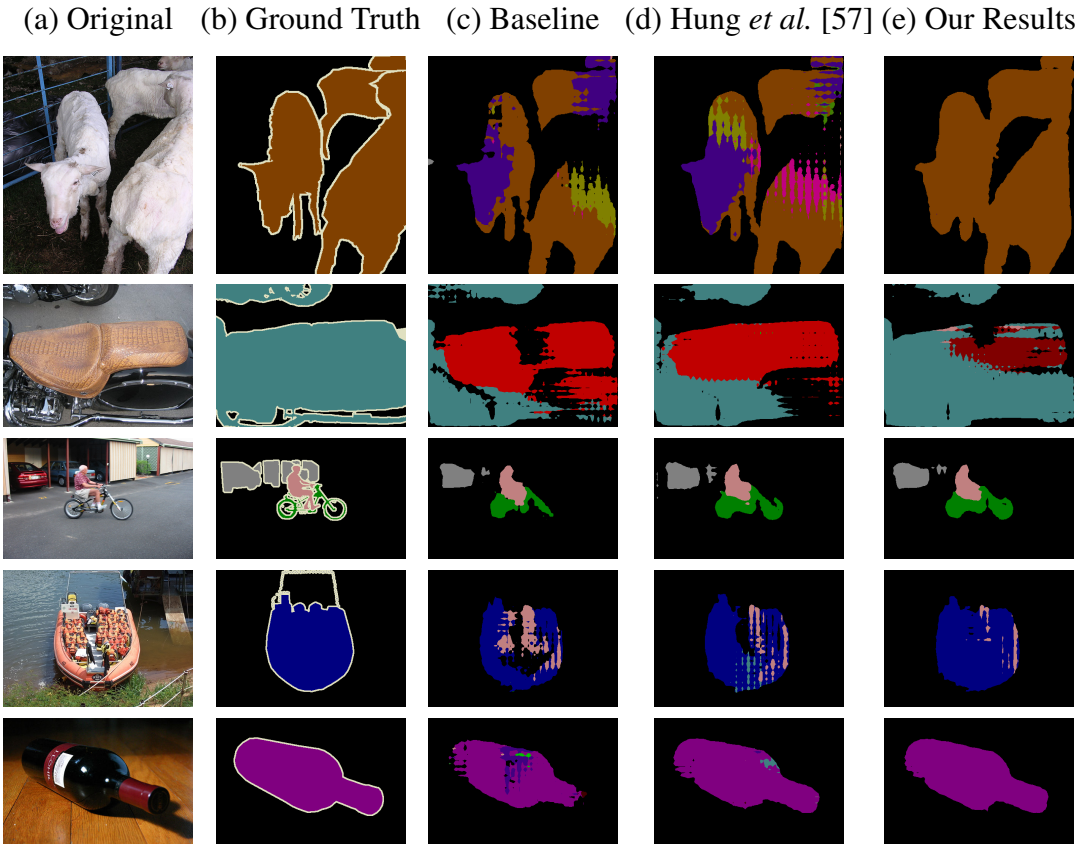


Fig. 2.4 Qualitative results obtained using our semi-supervised segmentation approach on the PASCAL VOC dataset with 5% labeled data without COCO pre-training.

Recent methods. Since our method was proposed, there have been many follow-up works [54, 22, 123, 132]. We also implemented our approach with new upgraded network settings proposed by these approaches and compared the upgraded performance of our model with these new baselines. Shared attributes among these new works include the usage of DeepLabv3+ architecture and the higher-resolution input images. These follow-up works are majorly built on our self-training idea along with new advancements. CPS [22] utilized two parallel trainable networks to generate pseudo labels for each other. Cutmix-seg [36] introduces the usage of CutMix augmentation into semantic segmentation using a mean-teacher model inspired by successful SSL image classification works [131, 110]. U2PL proposed to utilize unreliable pseudo-labels using contrastive learning along with reliable pseudo-labels using self-training. Many approaches [54] have been proposed to resolve class imbalance, which might be present in the dataset or can be exaggerated by the self-training approach. Lately, Xu *et al.* [132] proposed a prototype-based consistency regularization method to capture the intra-class variations. Our upgraded model uses Deeplabv3+ architecture with higher resolution input images and cutmix augmentation. In this work, we compare the

without COCO pre-training				
Methods	**Labeled Data**			
	1/50	1/20	1/8	Full
DeepLabv2	48.3	56.8	62.0	70.7
Hung *et al.*[57]	49.2	59.1	64.3	71.4
Ours (s4GAN only)	58.1	60.9	65.4	71.2
Ours (s4GAN + MLMT)	**60.4**	**62.9**	**67.3**	**73.2**
with COCO pre-training				
DeepLabv2	53.2	58.7	65.2	73.6
Hung *et al.*[57]	57.2	64.7	69.5	74.9
Ours (s4GAN only)	60.9	66.4	69.8	73.9
Ours (s4GAN + MLMT)	**63.3**	**67.2**	**71.4**	**75.6**

Table 2.1 Semi-supervised semantic segmentation results on the PASCAL VOC dataset without and with COCO pre-training. Our model is trained with input images of resolution 321×321.

Method	**Resolution**	**Labeled Data**		
		1/16 (662)	1/8 (1323)	1/4 (2646)
Baseline [22]	512x512	70.59	73.20	76.62
CPS [22]	512x512	74.48	76.44	77.68
AEL [54]	512x512	77.20	77.57	78.06
U2PL (w/ Cutmix) [123]	512x512	77.21	79.01	79.30
Proto-Cons [132]	512x512	78.60 (+8.01)	80.71 (+7.51)	80.78 (+4.16)
Baseline*	305x305	69.88	73.65	76.11
Ours	305x305	72.49	75.94	77.29 (+1.18)
Ours (w/ Cutmix)	305x305	74.21 (+4.33)	76.66 (+3.01)	77.02

Table 2.2 Semi-supervised semantic segmentation results on the PASCAL VOC dataset. Here, we compare our upgraded method to some of the latest methods. Our upgraded model is trained using DeepLabv3+ architecture, similar to other compared methods in the table. * refers to our implementation.

upgraded version of our model with several latest methods, including CPS, AEL, U2PL, and Proto-Cons.

2.4 Results

PASCAL-VOC Table 2.1 shows the segmentation results on the PASCAL VOC dataset with and without pre-training on the Microsoft COCO [77] dataset. We achieved improved results compared to the previous method for all data splits. Our method achieves a performance increase of 5% to 12% over the baseline for different data splits by utilizing unlabeled

samples without pre-training the network on any segmentation dataset. Notably, the approach works well even with only 2% (1/50) of labeled data. Figure 2.4 shows qualitatively how our method helps remove artifacts produced by other methods. We also validated our approach with COCO pre-training to directly compare with Hung *et al.* [57] and achieved an improvement of 6.1 mIoU points over them for the 1/50 split. We speculate that [57] is inferior in the low-data regime due to the two-stage GAN training, where the discriminator is only updated based on the labeled samples. This effectively reduces the amount of data it sees during training, which can easily lead to overfitting.

Comparison to new state-of-the-art Table 2.2 compares our method with the latest state-of-the-art methods. We upgrade our model to make use of a larger batch size of 16, similar to the latest methods. Latest works [132, 123] also use higher-resolution input images with a resolution of 512x512. However, we can only use 305x305 resolution images with a batch size of 16 due to limited GPU memory. We get bigger performance gains from the usage of large batch size compared to using higher resolution input images given a fixed memory usage. By utilizing a larger batch size, we obtain a much higher baseline performance of 69.88 mIoU compared to our previous baseline performance of $\sim$60 mIoU with $1/16$ labeled samples. Using the s4GAN branch, we obtain a consistent improvement over the baseline. Similar to these new methods, we also included the CutMix augmentation technique based on student-teacher modeling. We found that our methods improve significantly using such strong augmentation and show competitive performance compared to the latest methods.

Cityscapes On the Cityscapes dataset, the s4GAN branch yields an improvement over the baseline of 3.1%, and 1.7% for the 1/8 and 1/4 data splits, respectively; see Table 2.3. The distribution of different classes in this dataset is highly imbalanced. The vast majority of the classes are present in almost every image, and the few remaining classes occur only scarcely. In this situation, a classifier that eliminates labels of non-existing classes does not help. Thus, our MLMT branch was ineffective for the Cityscapes dataset.

Figure 2.5 shows qualitative results obtained using our approach with 1/8 labeled samples and the remaining unlabeled samples. The differences in the Cityscapes dataset are subtle. Therefore, we include the zoomed-in views of informative areas. Images from Figure 2.5 show our approach yields improvement over the baseline.

Comparison to new state-of-the-art. Similar to the results on PASCAL-VOC, we compare our method on the Cityscapes dataset with the latest state-of-the-art methods. For the Cityscapes dataset, we find that the resolution of the input image plays a big role. Although the latest works [132, 123] uses input images with a resolution of at least 769x769 with a batch size of 16, we could only manage to use an input image of resolution 609x609

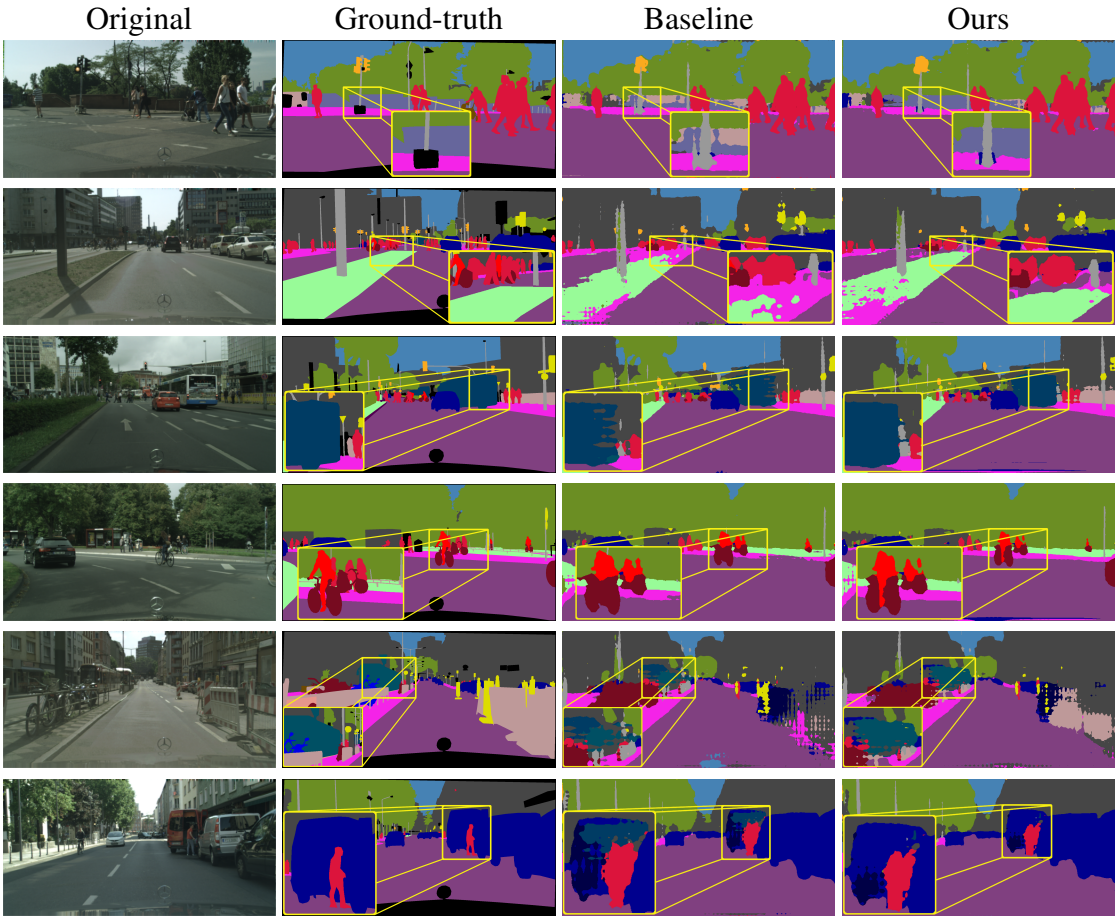


Fig. 2.5 Qualitative results on the Cityscapes dataset using 1/8 labeled samples without COCO pre-training. The proposed semi-supervised approach produces improved results compared to the baseline. We compare our ('Ours') results with the fully-supervised baseline ('Base') which is trained only on the labeled subset of data.

with a batch size of 8 due to memory limitations. Table 2.4 compares our proposed method s4GAN with previous approaches. We find that our approach delivers similar improvements compared to other latest approaches. Our approach improves over the 1/8 baseline by 2.96 points, whereas the best method improves by 3.95 points.

PASCAL-Context Our approach successfully generalizes to the whole scene parsing the PASCAL-Context dataset. Table 2.5 shows the performance on two splits (1/8 and 1/4 labeled data) of PASCAL-Context. Although this dataset is smaller and more difficult than PASCAL VOC, there is still an improvement over the baseline of 3.2% and 2.4% for the 1/8 and 1/4 splits, respectively. Figure 2.6 show qualitative results on the PASCAL-Context test set using 1/8 labeled samples and the remaining unlabeled samples. PASCAL-Context is a smaller

Method	Labeled Data		
	1/8	1/4	Full
DeepLabv2	56.2	60.2	66.0
Hung *et al.*[57]	57.1	60.5	**66.2**
Ours (s4GAN only)	**59.3**	**61.9**	65.8

Table 2.3 Semi-supervised semantic segmentation results on the Cityscapes dataset without COCO pre-training.

Method	**Resolution**	**Labeled Data**		
		1/8 (372)	1/4 (744)	1/2 (1488)
Baseline [123]	769x769	72.53	74.43	77.83
CPS [22]	769x769	74.31 (+1.78)	74.58 (+0.15)	76.81 (-1.02)
AEL [54]	769x769	75.55 (+3.02)	77.48 (+3.05)	79.01 (+1.18)
U2PL (w/ Cutmix) [123]	769x769	74.37 (+1.84)	76.47 (+2.04)	79.05 (+1.22)
U2PL (w/ AEL) [123]	769x769	76.48 (+3.95)	78.51 (+4.08)	79.12 (+1.29)
Proto-Cons [132]	769x769	76.31 (+3.78)	78.40 (+3.97)	79.11 (+1.28)
Baseline*	609x609	70.25	71.54	74.39
Ours	609x609	73.21 (+2.96)	75.59(+4.05)	75.41(+1.02)

Table 2.4 Semi-supervised semantic segmentation results on the Cityscapes dataset compared to the latest state-of-the-art works with the high-resolution input image. Our method is trained on comparatively lower input resolution of 609 $\times$ 609 due to memory limitations. * refers to our implementation

and harder dataset as compared to PASCAL VOC. Therefore, the results are not as visually appealing. Still, there is a clear improvement over the baseline.

2.4.1 Ablation study

All the experiments for the ablation studies are shown on the PASCAL VOC dataset without COCO pre-training.

Contribution of the two branches. Table 2.6 shows the contribution of the s4GAN branch and the MLMT branch. The s4GAN branch is able to extract extra dense information using unlabeled images. It improves the shape of the segmented objects, makes the segmentation prediction more coherent by filling small holes, and improves the fine boundaries between the foreground and background. We qualitatively showcase these improvements in Figure 2.9(e).

The MLMT branch plays a complementary role and removes the false positives from the predictions. Figure 2.9(d) shows the improvement using the 'MLMT branch only' with the

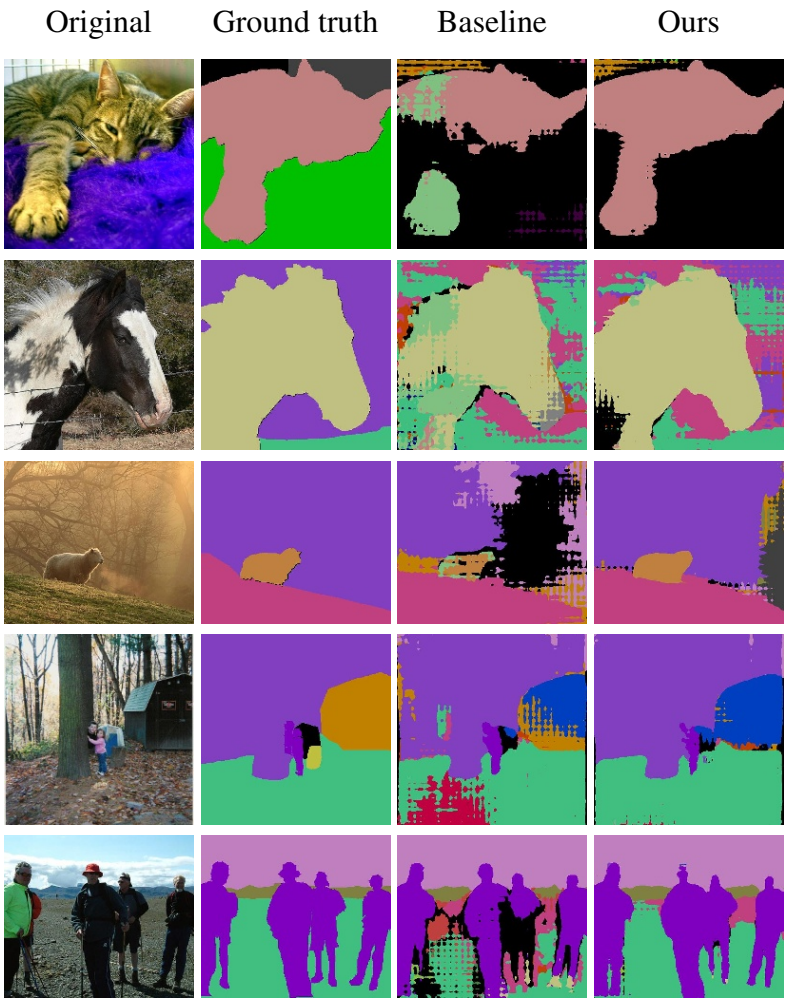


Fig. 2.6 Qualitative results on the PASCAL-Context dataset using 1/8 labeled samples. Our approach produces improved results compared to the baseline. We compare our ('Ours') results with the fully-supervised baseline, which is trained only on the labeled subset of data.

segmentation baseline method, and Figure 2.9(g) shows the improvement using the MLMT branch together with the s4GAN branch. The MLMT branch makes use of unlabeled images to extract image-level information about the presence of certain classes in the image. For some cases, the s4GAN branch introduces new artifacts, which are also filtered out by the MLMT branch. This effect is shown in the bottom-row example of Figure 2.9.

Different s4GAN branch loss terms. We trained the generator network with a combination of the cross-entropy (CE) loss, the feature matching (FM) loss, and the self-training (ST) loss. To justify this configuration, we compare the system performance when using different loss terms; see Table 2.8. There is a consistent performance increase when adding all the proposed loss terms. We found it crucial for the system stability to train using the FM loss and not the

Method	Labeled Data		
	1/8	1/4	Full
DeepLabv2	32.1	35.4	41.0
Hung *et al.*[57]	32.8	34.8	39.1
Ours (s4GAN only)	34.4	37.1	40.8
Ours (s4GAN + MLMT)	**35.3**	**37.8**	**41.1**

Table 2.5 Semi-supervised semantic segmentation results on the PASCAL-Context dataset without COCO pre-training.

Method	Data		mIoU
	labeled(%)	unlabeled(%)	
DeepLabv2	5	None	56.8
s4GAN only	5	95	60.9
MLMT only	5	95	59.0
s4GAN + Threshold	5	95	61.2
s4GAN + Class-wise Threshold	5	95	61.5
s4GAN + CNN	5	95	62.2
s4GAN + MLMT	5	95	**62.9**

Table 2.6 Ablation study of the contribution of each branch. Results are shown for the 5:95 data split on the PASCAL VOC dataset.

standard GAN loss. Figure 2.7 illustrates the effect of using our proposed self-training loss. We plot how the discriminator score changes during the course of training. The scores are averaged over 100 iterations of fake (generated) and real (ground-truth) samples separately. As discussed in Sec. 2.2.1, adding the ST loss impedes the progress of the discriminator and does not allow it to become overly confident; that is, it draws its predicted scores towards 0.5. This has a positive effect on the generator performance, in particular with few labeled samples, as can be seen from the last line of Table 2.8.

Semi-supervised multi-label classification. In this experiment, we compared the performance of the proposed MLMT branch with a standard supervised classifier. Table 2.6 shows that we already get an improvement of 1.3% over the s4GAN performance just by using a CNN-based classifier [49], but when we further add the consistency-based semi-supervised classification approach, we observe that the performance improvement increases to 2%. We also conducted a simple heuristic experiment where we deactivated the predicted class channels which have a pixel count less than a threshold. In Table 2.6, 's4GAN + Threshold' refers to the case where a single threshold is set for all the classes, and 's4GAN + Class-wise

Method	**Labeled Data**			
	1/50	1/20	1/8	Full
Deeplabv2 (v2)	48.3	56.8	62.0	70.7
Ours v2 (s4GAN + MLMT)	60.4	62.9	67.3	73.2
Deeplabv3+ (v3+)	unstable	unstable	63.5	74.6
Ours v3+ (s4GAN + MLMT)	62.6	66.6	70.4	74.7

Table 2.7 Results on PASCAL VOC without COCO pre-training using different backbone architectures.

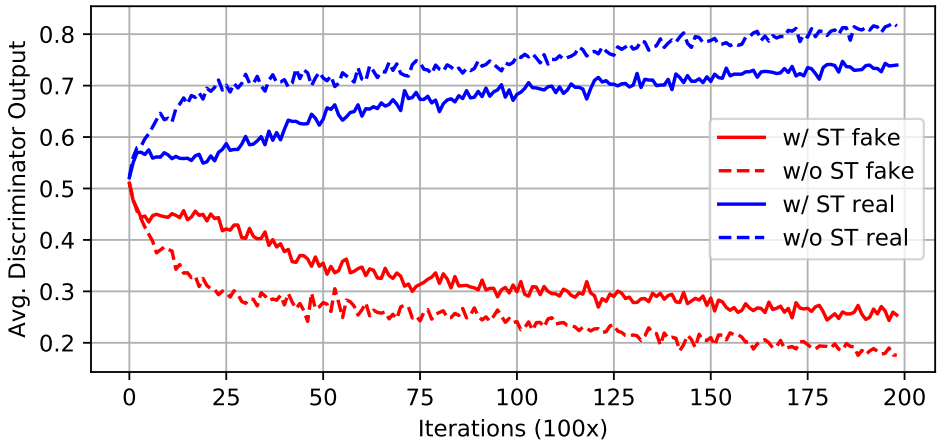
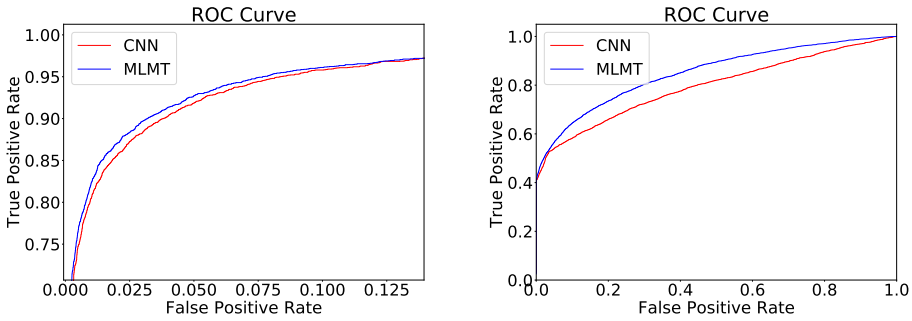


Fig. 2.7 Evolution of the discriminator output during the course of training averaged over real and fake samples separately. Using the self-training loss (w/ ST) prevents D from becoming overly strong and results in better training dynamics compared to the case when self-training is disabled (w/o ST).

Threshold' refers to the case where each class has its best respective threshold. We search for the best-performing thresholds on the validation set in the range from 1K to 12K pixels at an increment step of 1K. Figure 2.9(f) and (g) show the effect of adding a CNN-based classifier and an MT-based semi-supervised classifier, respectively.

We also analyze the performance of the CNN-based multi-label classification and MLMT-based semi-supervised multi-label classification independent of the segmentation model. Figure 2.8 shows the comparison between the ROC curves of the two methods on the task of multi-label classification. The MLMT classifier obtains a lower false positive rate for the same true positive rate. The effect is even more pronounced when not using ImageNet pre-training; see Figure 2.8(b). This mode of operation is important for domains where ImageNet pre-training does not help, e.g., bio-medical image analysis.

Limitations In certain situations, our method produces imprecise predictions. Sometimes object classes with multiple protrusions, like plant leaves, chair legs, etc., are under-



(a) (b)

Fig. 2.8 ROC curves for CNN-based classification and MT-based semi-supervised classification method using 5% labeled data with (a) and without (b) ImageNet pre-training. MT produces fewer false positives, especially when training from scratch.

Loss Terms	**Labeled Data**		
	1/50	1/20	1/8
CE only	48.3	56.8	62.0
CE + SGAN [46]	54.0	57.1	62.5
CE + FM	55.4	58.4	63.9
CE + FM + ST	**58.1**	**60.9**	**65.4**

Table 2.8 Ablation study of different GAN loss terms for the generator on the PASCAL VOC dataset. SGAN refers to the standard GAN loss [46], FM refers to the feature-matching loss and ST refers to the self-training loss.

segmented by the s4GAN branch, as shown in Figure 2.10(first row). Occasionally, our approach can identify certain ambiguous foreground objects as one of the classes, as shown in Figure 2.10(second row). Also, there exist a few cases where some truly positive results are wrongly predicted by the classifier. However, both qualitative and quantitative results confirm that these failure cases are outweighed by the positive effect of the proposed techniques. In Figure 2.10 (row 3-4), we include a few failure cases for the PASCAL-context dataset using our approach. Figure 2.11 shows a few failure cases for the Cityscapes dataset where a few thin objects were not segmented properly using our approach.

2.4.2 Semi-supervised Semantic Segmentation with Weak-labels

To further validate the effectiveness of our approach, we compare it to other semi-supervised segmentation methods [94, 125] that utilize extra weak image-level annotations - labels of classes present in the image. In this weakly semi-supervised setting, class labels are provided as extra supervision for all the images, along with full pixel-wise labels for a few images during training. Here, we compare the performance of our approach with methods that use

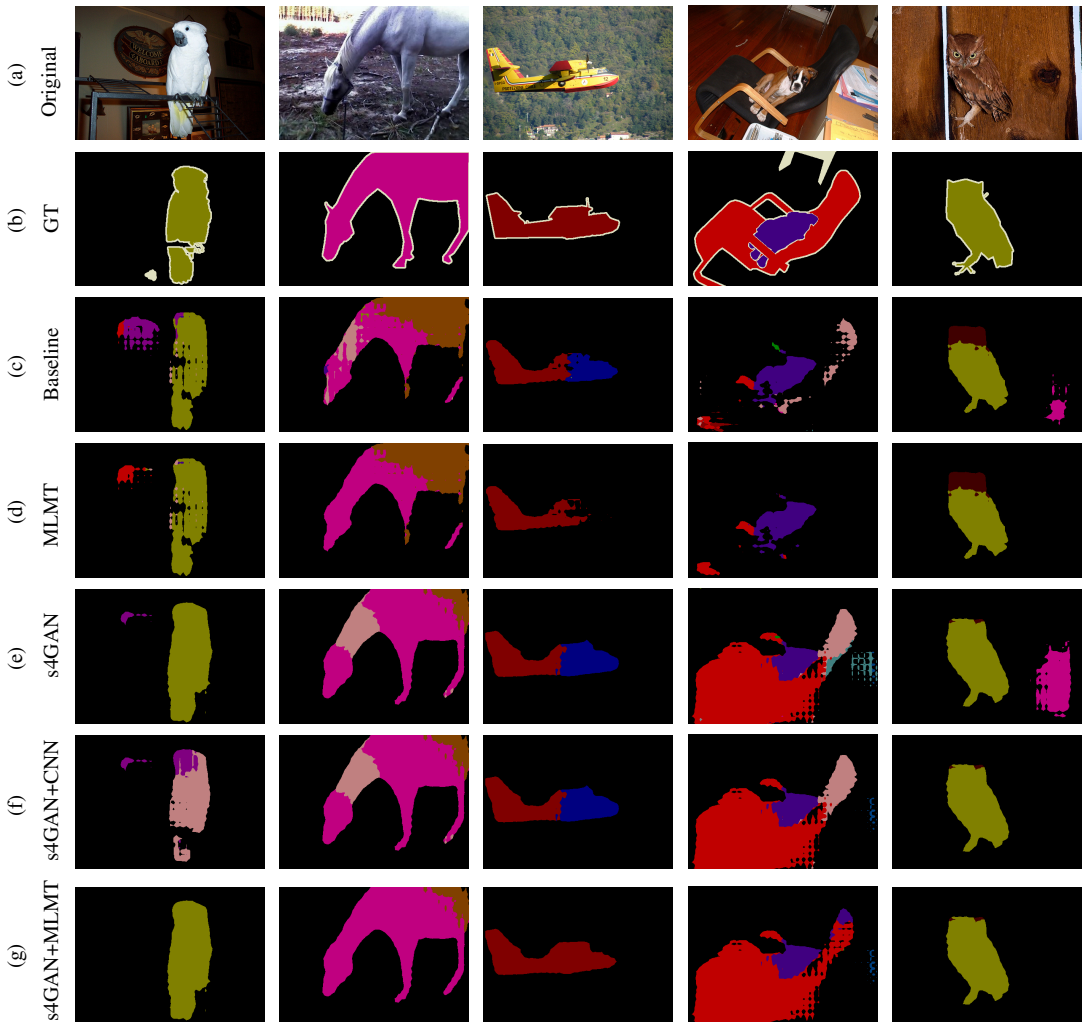


Fig. 2.9 Ablation study on the PASCAL VOC dataset showing the contribution of the MLMT (d) and the s4GAN (e) branches individually. The s4GAN and the MLMT branches together show a complementary behaviour fixing both low and high-level artifacts (g). These results are obtained using 5% labeled data.

extra image-level annotations *i.e.*, 1,464 strongly (w/ segmentation masks) annotated images from the original PASCAL VOC dataset and 9,118 weakly (image-level) annotated images from the augmented SBD dataset. To use extra image-level annotations, we train the MLMT branch using extra image-level labels for improved multi-label classification. The training procedure and hyperparameters remain exactly the same as in the previous semi-supervised setting. Table 2.9 summarizes the semi-supervised semantic segmentation results with extra $\sim$9K image-level annotations. We achieve an improvement of 5.2% over the baseline. Unlike previous methods, our approach does not utilize CRF post-processing.

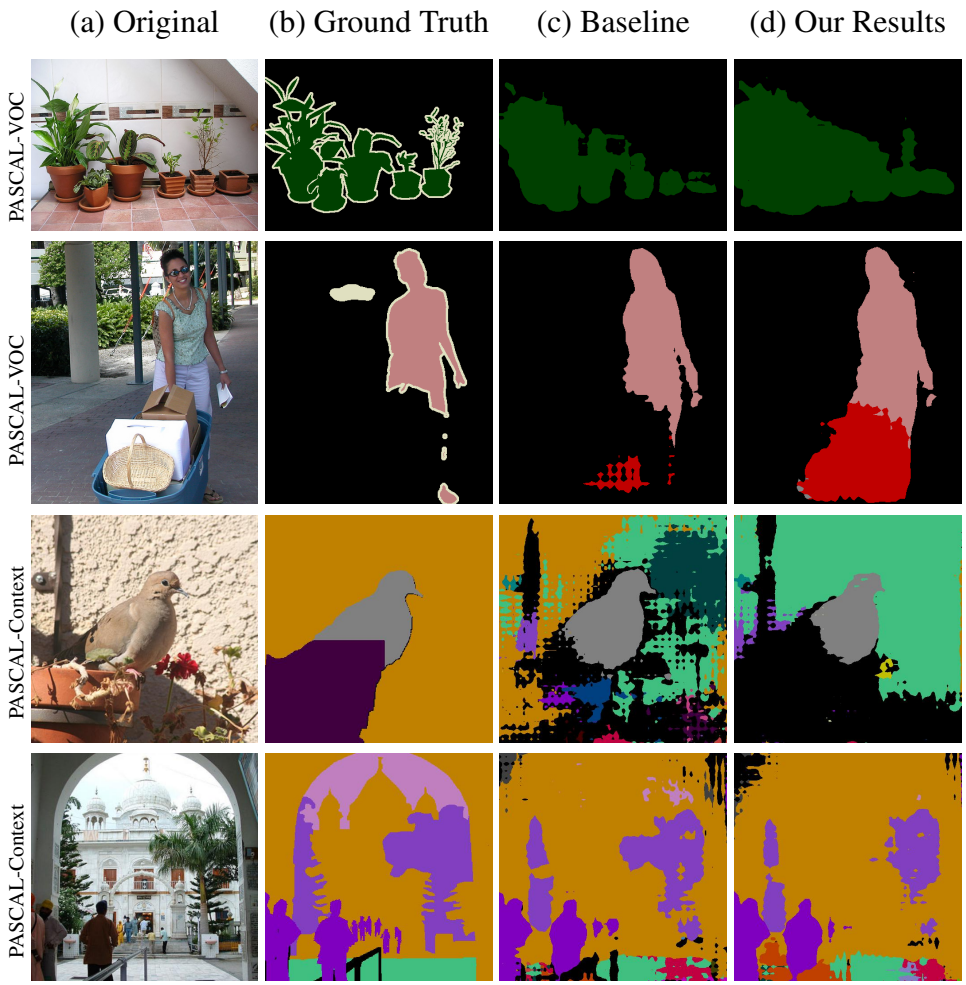


Fig. 2.10 Failure cases. Qualitative results on the PASCAL-VOC and PASCAL-Context datasets using 1/8 labeled samples. Failures of our approach. We compare our ('Ours') results with the fully-supervised baseline ('Base'), which is trained only on the labeled subset of data.

In an additional experiment, we show that our approach can also make use of extra weak supervision based on random scribbles. These random scribbles are freestyle hand-drawn lines annotating the pixels belonging to a particular class. For the experiments, we provide one scribble of each class instance in an image for the unannotated set of images in the semi-supervised learning setting. We use the PASCAL-scribble [75] dataset for our experiments. Table 2.10 shows that using additional scribble annotations in the s4GAN approach improves the model performance by 7.1% mIoU and 5.3% mIoU for the 2% and 5% cases, respectively. This experiment shows the extra flexibility of the proposed model in handling different types of supervision.

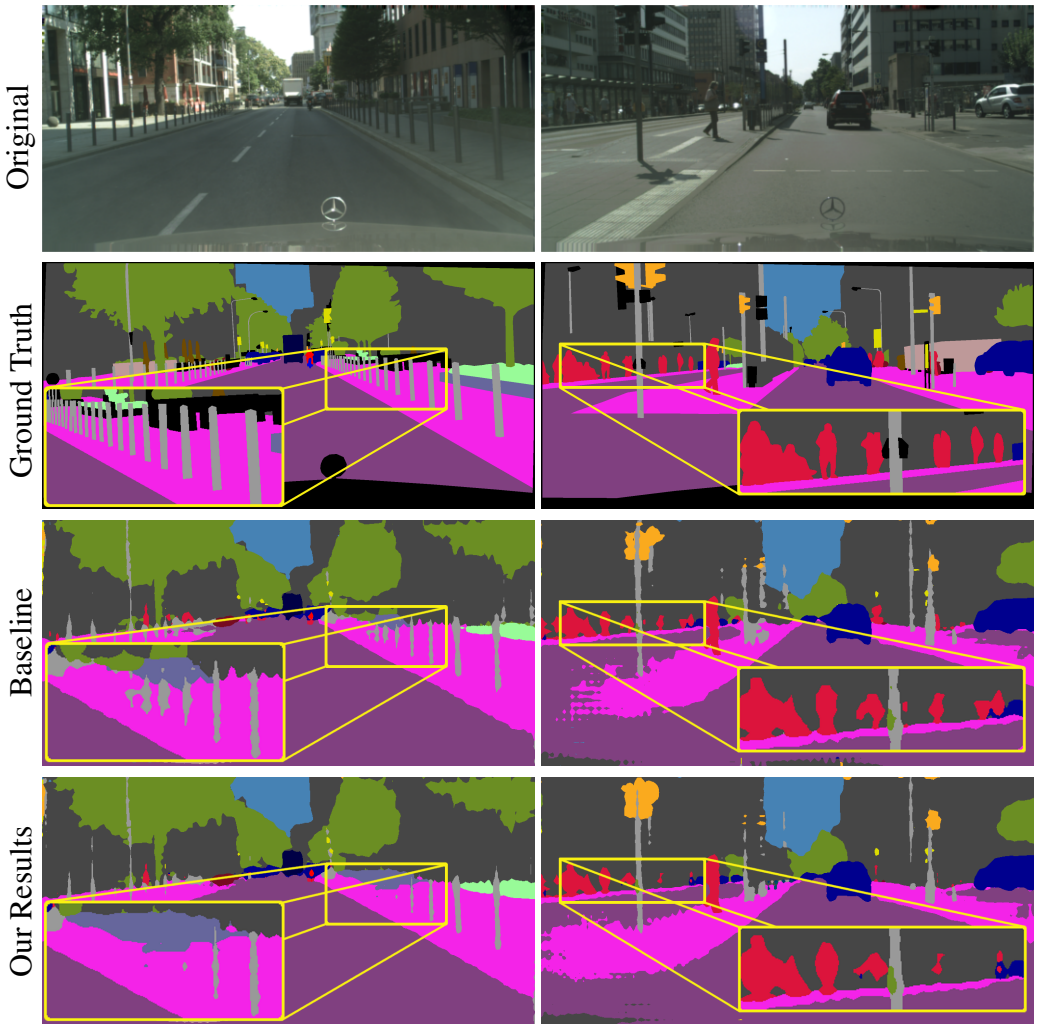


Fig. 2.11 Failure cases. Qualitative results on the Cityscapes dataset using 1/8 labeled samples. Failures of our approach. We compare our ('Ours') results with the fully-supervised baseline ('Base') which is trained only on the labeled subset of data.

2.5 Summary

In this chapter, we presented a two-branch approach to the task of semi-supervised semantic segmentation. The proposed branches are designed to alleviate both low-level and high-level artifacts, which often occur when working in a low-data regime. The two branches complement each other, which is validated both qualitatively and quantitatively in this work. Our proposed feature matching loss serves to stabilize the adversarial training process even in scenarios with insufficient labeled data, while the self-training loss improves the balance between the GAN generator and discriminator, thus increasing the final performance. Overall, our approach is quite versatile that can be adapted to different types of supervision like image-

Method	Data Split (Strong/Weak/Unlab)		
	1.4K/0/9K	1.4K/9K/0	All/0/0
DeepLab-CRF-LargeFOV [18]	62.5	—	67.6
WSSL (CRF) [94]	—	64.6	—
MDC [125]	—	62.7	—
MDC (CRF) [125]	—	65.7	—
DeepLabv2	65.7	—	70.7
Ours (s4GAN only)	67.5	—	71.2
Ours (s4GAN + MLMT)	69.6	**70.9**	73.2

Table 2.9 Semi-supervised semantic segmentation results on the PASCAL VOC dataset using extra weak image-level annotations. Data splits (*A/B/C*) refers to the usage of *A* pixel-wise labeled samples, *B* image-level labeled samples and *C* unlabeled samples. For *e. g.* the second column heading shows the case where 1.4K images are provided with pixel-wise class labels and 9K images are provided with only image class labels.

Method	Labeled data		
	2% (210)	5%(525)	Full(10k)
DeepLabv2	48.3	56.8	70.7
Ours (s4GAN only)	58.1	60.9	71.2
with additional scribbles			
DeepLabv2	61.5	62.8	—
Ours (s4GAN only)	65.2	66.2	—
Ours (s4GAN + COCO pre-train)	68.1	69.7	73.6

Table 2.10 Semi-supervised semantic segmentation results with extra weak supervision using scribbles on the PASCAL VOC dataset without and with COCO pre-training.

wise class labels and scribbles. The effectiveness of this design is demonstrated in a series of extensive experiments on three standard segmentation benchmarks.

Several components proposed in this work are still used by the latest works, including the online self-training technique and multi-label classification branch. Most of these methods merge the multi-label classification branch into the segmentation branch by using an additional network head. After studying our approach and the latest methods, we find that the self-training approach using pseudo-labeled is a crucial component that helps in learning from unlabeled samples. It is a common component in many state-of-the-art methods. Usage of strong augmentation, for *e. g.* Cutmix, Cutout, is also ubiquitous to most of these methods using a student-teacher model. Class-balanced training can be an important component when the dataset is biased, for *e. g.* Cityscapes dataset. Some additional components, like the use

of low-confidence pseudo labels using contrastive learning and consistency regularization towards intra-class prototypes, can further boost performance.

In this chapter, we proposed a new semi-supervised method for semantic segmentation to learn a label-efficient model. In semi-supervised learning setting, a small set of labeled samples are assumed to be given, randomly picked in this case for experimental purposes. However, these samples can also be intelligently selected for annotation from the large unlabeled pool of samples. Such informed selection might allow us to select more valuable samples for annotation and achieve better performance, keeping the annotation cost constant. This task is referred to as active learning in the literature and is a non-trivial task that requires an understanding of the underlying data distribution, implicit bias in the model representation, and the task objective. In the next chapter, we study active learning methods for deep neural networks for different tasks. We investigate several aspects of deep active learning, including its effectiveness, correct evaluation procedure, and best usage in different scenarios.

Chapter 3

Realistic Deep Active Learning

The content of this chapter was adapted from the following papers.

> Sudhanshu Mittal, Maxim Tatarchenko, Ozgun Cicek and Thomas Brox. Parting with Illusions about Deep Active Learning. In ArXiv 2019.

> Sudhanshu Mittal*, Joshua Niemeijer*, Jörg Schäfer and Thomas Brox (*indicates equal contribution). Best Practices in Active Learning for Semantic Segmentation. DAGM German Conference on Pattern Recognition 2023.

Joshua Niemeijer is a co-author with significant contributions to the paper "Best Practices in Active Learning for Semantic Segmentation." He identified the need for a more realistic evaluation of Active Learning methods in the context of autonomous driving, as the previous SOTA focused on datasets curated for diversity. He proposed and co-designed the study on scenarios closer to the redundant measurement campaign data found in real-life scenarios. Joshua contributed by implementing the BALD algorithm for semantic segmentation. He performed baseline experiments on the Cityscapes dataset and curated redundant dataset pools from the original A2D2 dataset. All co-authors actively participated in the project discussions as well as the final paper text editing. All the other contributions described in this chapter are made by myself.

* * *

In Chapter 2, we proposed a semi-supervised method for semantic segmentation that could learn from a very small set of labeled samples along with a large set of unlabeled samples. In that setting, we learned from a set of labeled samples that were already given. In this chapter, we study whether it is possible to select better samples for annotation than just random selection to achieve better model performance from the same amount of annotation.

This process of selecting samples for human annotation and training the model using these annotated samples is referred to as Active Learning. The objective of this chapter is to provide a realistic assessment concerning the relevance and effectiveness of deep active learning for various vision tasks.

In *Active Learning* (AL), the objective is the reduction of annotation cost by selecting those samples for annotation, which are expected to yield the largest increase in the model's performance. Active learning is based on the attractive idea that some samples are more valuable for learning than others - by identifying those in the pool of unlabeled data, we can use an annotator's time more efficiently. It assumes that raw data can be collected in abundance for most large-scale data applications, but annotation limits the use of this data.

Problem statement. Assuming we have a dataset consisting of n examples, denoted as $\mathcal{D} = \mathbf{x}, \mathbf{y}$, where $\mathbf{x}$ is the set of input images, and $\mathbf{y}$ is the set of corresponding labels. Let $\mathcal{D}^{\ell} = \mathbf{x}^{\ell}, \mathbf{y}^{\ell}$ denote the set of labeled examples, and $\mathcal{D}^{u} = \mathbf{x}^{u}$ denote the set of unlabeled examples. Thus $\mathcal{D} = \mathcal{D}^{\ell} \cup \mathcal{D}^{u}$, where $\mathcal{D}^{\ell}$ contains $\mathcal{B}_i$ samples according to the initial labeling budget, and $\mathcal{D}^{u}$ contains $n - \mathcal{B}_i$ samples.

The objective of active learning is to learn a function $f : \mathcal{X} \rightarrow \mathcal{Y}$ that maps input images $\mathbf{x} \in \mathcal{X}$ to labels $\mathbf{y} \in \mathcal{Y}$, where $\mathcal{X}$ is the space of images, and $\mathcal{Y}$ is the space of labels. The function f is learned in cycles. At each cycle c, the active learning method selects a set of $\mathcal{B}_s$ samples according to the sampling budget from $\mathcal{D}^{u}$ for annotation using a query function, also called the acquisition function. The acquisition function is a scoring function that identifies the most valuable samples for selection. Most valuable samples could mean samples that are most uncertain, most diverse, or most representative of the data distribution. The acquisition function defines the identity of the active learning method being used. After, the selection, the model function f_c is then trained using the current labeled set $\mathcal{D}^{\ell}_{c} = \mathcal{D}^{\ell}_{c-1} \cup \mathcal{B}_s$ and current unlabeled set $\mathcal{D}^{u}_{c} = \mathcal{D}^{u}_{c-1} \setminus \mathcal{B}_s$.

$$f_c = \arg\min_{f} \; \mathcal{L}(f; \mathcal{D}^{\ell}_{c}, \mathcal{D}^{u}_{c}) \tag{3.1}$$

The active learning cycle is repeated either until the maximum annotation budget is exhausted or the desired performance level is reached. In summary, the active learning methods iteratively select a subset of unlabeled samples to be annotated by an oracle and trains the model on this updated set, with the objective of maximizing the performance on a separate test set. In this work, for both image classification and semantic segmentation tasks, we assume that the cost of annotation per image is equal across the dataset. This assumption is made based on our empirical study shown in Chapter 3.3.5.

Motivation. The appeal of the active learning idea has spawned a multitude of ConvNet-based AL methods. Various previous works have proposed solutions to this challenge, which is ubiquitous in most machine learning applications. Yet there exists a skepticism amongst the users, whether it brings any additional benefit over selecting the samples randomly or based on some manual prior. One of the main reasons for such hesitancy is the inconsistency of the method performances across published works due to incompatible evaluation settings like different architectures, augmentation strategies, optimization methods, etc. How these acquisition methods perform w.r.t the difficulty levels of the task and underlying data distributions is also rather unknown. Semi-supervised learning, besides active learning, is a way to deal with this situation of high annotation cost, as studied in Chapter 2. Semi-supervised learning (SSL) and AL share a common objective of obtaining maximum performance from minimum supervision. Therefore, it is sensible to integrate both ideas, yet the combination of active learning with semi-supervised learning is understudied. In this chapter, we aim to study this combination in detail and provide clarity on the above-mentioned discrepancies.

Scope of this chapter. In this chapter, we objectively assess the state of the field and challenge the principal hypothesis behind active learning: *active selection of the samples to be labeled leads to a significant reduction in the annotation effort compared to random selection.* We systematically study the behavior of active learning methods under different training conditions in order to present a realistic perspective. Our study identifies that existing works are effective, but only under certain training conditions. They are not consistent across different model variabilities like data distribution, annotation budget, supervision type, and regularization. This chapter provides an extensive analysis of existing active learning methods under these diverse variabilities for both image classification and semantic segmentation tasks.

We conduct a detailed analysis in two parts. The first part (Chapter 3.2) studies the nature of AL methods for image classification, and the second part (Chapter 3.3) studies the nature of AL methods for semantic segmentation. For the image classification task, we first challenge the existing methods across several similar datasets like CIFAR-10 and CIFAR-100 to check the consistency of the methods. Then, they are subsequently studied under the influence of strong augmentations, semi-supervised learning objectives, and, lastly, under different annotations budgets. For the dense semantic segmentation task, data is often collected as video streams, especially for navigation applications such as autonomous driving. Such video stream data is very different from previously tested benchmarks in active learning literature; it is highly redundant. Therefore, the behavior of AL methods w.r.t. such data

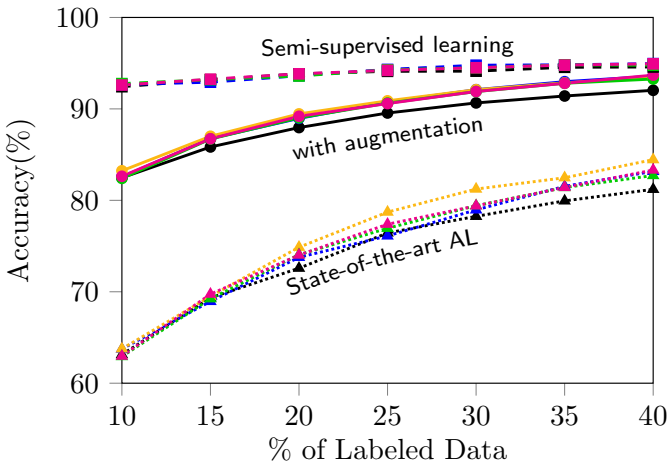


Fig. 3.1 State-of-the-art active learning methods do not consistently use modern data augmentation techniques or advances in the closely related field of semi-supervised learning, which leads to the wrong impression about the current state of the field. Results are shown for image classification on CIFAR 10.

distribution changes is unknown and highly relevant. To understand the nature of active learning methods under such missing cases, we evaluate methods for the segmentation task for datasets with different levels of redundancy. We also study the active learning methods for semantic segmentation across different annotation budget settings and with the integration of semi-supervised learning.

Active Learning for Image Classification

Our first study seeks answers to the following scientific questions about active learning for image classification:

- Since a widely accepted evaluation protocol is missing, methods are often tested under incompatible circumstances: different architectures, different augmentation strategies, etc. We evaluate the effect of compatible experimental settings on the ranking of methods. In particular, **do AL methods work consistently well in conjunction with data augmentation?**

- Contemporary papers on active learning largely ignore the progress of the closely related field of semi-supervised learning, where approaches effectively operate under the same assumptions with regard to the used data. **What is the effect as concepts from semi-supervised learning are integrated into active learning?**

- Existing methods are typically not evaluated in a low-budget setting - a mode crucially important to kick-start network training on a new dataset. **How do active learning concepts work in such a low-budget regime?**

Keeping in mind the aforementioned questions, we perform an extensive comparison of existing approaches for image classification. We study 5 existing active learning acquisition methods across three dimensions - subject to regularization techniques like strong data augmentation, integration of semi-supervised learning, and under low as well as large annotation budget settings. Our experiments reveal that the progress recently made in the field of active learning is practically negligible when viewed under more realistic circumstances: in particular, using modern data augmentation and taking the advances of semi-supervised learning into account, see Figure 3.1. Integration of modern semi-supervised learning into active learning gives a significant boost to the acquisition functions. However, the difference w.r.t. random baseline with SSL becomes negligible. Active learning methods also fail to outperform simple random sampling, especially with a small labeling budget. Based on our extensive study, we suggest a more suitable evaluation protocol.

Active Learning for Semantic Segmentation

In the second part of the study, we conduct further analysis in this direction for the task of semantic segmentation. The semantic segmentation application opens up new dimensions for the analysis of active learning methods. As a result, we show that the findings for image classification only hold under certain conditions for semantic segmentation.

We noticed that the state-of-the-art active learning methods for segmentation had been evaluated only in a particular experimental setup - highly diverse benchmark datasets with a comparatively large annotation budget; see Table 3.1. Its applicability in other settings with different data distribution and annotation budgets is highly relevant but an unstudied topic. Additionally, we do not know how active learning methods integrate with semi-supervised learning. In this chapter, we also seek answers to specific missing questions not captured by previous works for the semantic segmentation task.

1. How do different active learning methods perform when the dataset has many redundant samples? Samples with highly overlapping information are referred to as redundant samples, for example, the consecutive frames of a video. Many commonly used segmentation datasets were originally collected as videos for practical reasons, e.g., Cityscapes, CamVid, BDD100k [137]. Since active learning methods were only tested on filtered versions of these datasets, their applicability on redundant datasets is open and highly

Dataset↓	Annotation Budget			
	Low		High	
Supervision →	AL	SSL-AL	AL	SSL-AL
Diverse	✓	✓	✓	✓
Redundant	✓	✓	✓	✓

Table 3.1 We study current active learning (AL) methods for semantic segmentation over 3 dimensions - dataset distribution, annotation budget, and integration of semi-supervised learning (SSL-AL). Green cells denote newly studied settings in this work. Previous AL works correspond to the grey cells. This work provides a guide to use AL under all the above conditions.

relevant.

2. What happens when the initial unlabeled pool is also used for training along with annotated samples using semi-supervised learning (SSL)? For image classification, many works [85, 41, 56, 91] including our work, have shown that integration of SSL into AL is advantageous. Most of these works appeared after our study on image classification. For semantic segmentation, this combination is not well studied.

3. What happens when the annotation budget is low? Which methods scale best in such low-budget settings? Semantic segmentation annotations can be expensive for specific applications, especially in the medical domain. Therefore, it is critical to understand the behavior of the various active learning methods in low-budget settings.

In this work, we report the results of an empirical study designed to find answers to the above-raised questions. We study 5 existing active learning methods across the three dimensions as mentioned above - subject to different data distributions w.r.t. redundancy in the dataset, including the integration of semi-supervised learning, and under low as well as large annotation budget settings, as shown in Table 3.1. The outcome of this study yields new insights and provides, as the major contribution of this work, a guideline for the best selection of available techniques under the various tested conditions. Figure 3.2 illustrates some of the results, particularly that the performance of acquisition functions can change depending on whether the dataset is redundant or diverse and that SSL integration plays an additional role in this. We observe that the integration of SSL and AL objectives can significantly improve model performance. However, the selection of an active learning method for segmentation is critical since a poor selection can have a detrimental impact when combined with SSL. Additionally, we show that active learning in a low annotation budget setting can be particularly volatile, even nullifying the complete need for it in some cases. This further emphasizes the importance of knowing the underlying data distribution.

We also propose a new exemplary evaluation task (A2D2-3K) for driving scenarios based on the highly redundant A2D2 dataset, which is closer to the raw data collection scheme in a

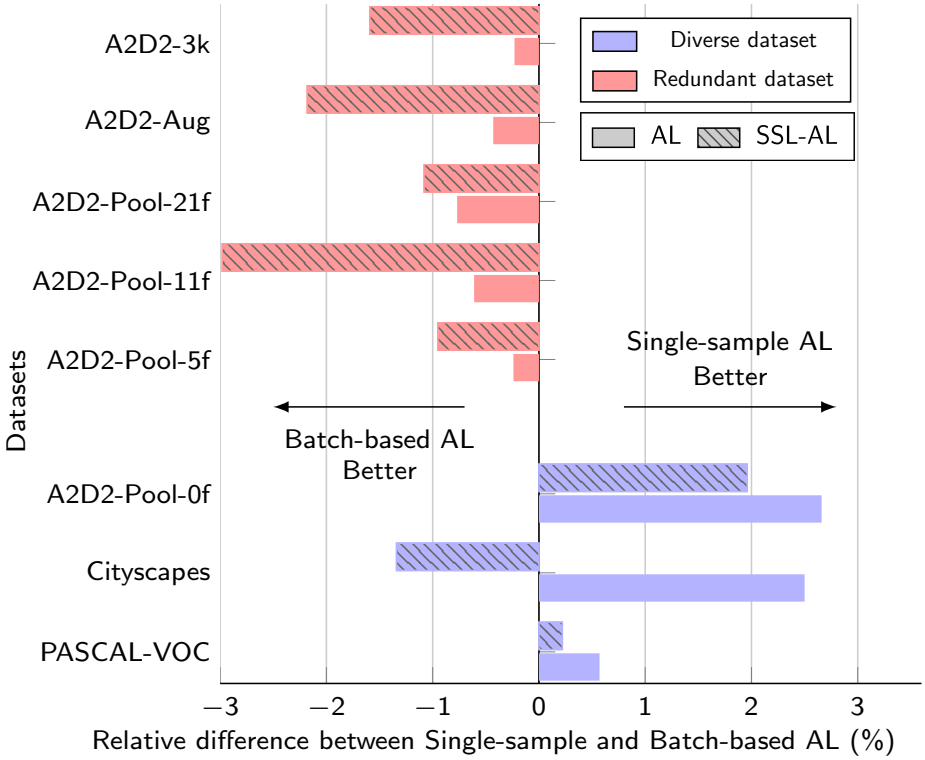


Fig. 3.2 We analyse and compare single-sample-based AL and batch-based AL on datasets with different levels of redundancy. The figure shows the difference between the best performing single-sample-based AL method and best performing batch-based AL method. We find that batch acquisition performs better for redundant datasets and single-sample acquisition performs better for diverse datasets. The integration of semi-supervised learning with active learning (SSL-AL) performs well for batch-based acquisition.

driving case. The experiment outcome on this task aligns with the findings of our study for redundant dataset type with a high annotation budget setting and shows that there is a strong case for using active learning in this context.

3.1 Related Work

Active learning methods can be categorized in multiple ways based on the nature of their heuristics. Different works indicate uncertainty is the right measure, whereas other works select samples based on their representativeness. Some works acquire a set by individually selecting the samples based on the sample value, and others acquire the whole batch with a cumulative objective.

3.1.1 Acquisition Objective: Uncertainty vs. Representation

Here, we discuss the works divided by the measure of the value of a sample.

Uncertainty-based methods try to find the samples which are hard to learn. In these methods, samples with the least predictive uncertainty are considered as most informative for labeling purposes. Several methods have been proposed to estimate uncertainty for neural networks using Bayesian [12, 38, 39, 63] and non-Bayesian approaches [71, 93]. Gal *et al.* [40] proposed to estimate posterior uncertainty using Monte Carlo dropout for active learning. Wang *et al.* [120] used the entropy of the softmax output in a neural network as a proxy uncertainty measure. Beluch *et al.* [11] used the ensemble method to estimate prediction uncertainty and select new samples based on a statistical measure of committee disagreement called variation ratio [62]. They show this method outperforms all other uncertainty-based methods.

Representation-based methods [103, 135], also referred to as density-based methods, try to find a diverse set of samples that optimally represents the complete dataset distribution. Sener *et al.* [103] formulated the active learning problem as core-set selection and showed effectiveness for CNNs. This method utilizes the geometry of data points using Euclidean distances and selects samples that maximize the coverage of all samples. *Learning-based approaches* [109, 136] use an auxiliary network module and loss function to learn a measure of information gain from new samples. Yoo *et al.* [136] proposed to learn a loss prediction module to predict target losses of unlabeled samples and select samples with the highest predicted loss. It can also be considered a pseudo-uncertainty heuristic. Sinha *et al.* [109] proposed a semi-supervised active learning approach that learns a VAE-GAN hybrid network to select unlabeled samples that are not well represented in the labeled set. It can also be considered a representation-type method.

3.1.2 Acquisition Type: Single-sample vs. Batch Acquisition

The acquisition methods can be categorized into single-sample-based and batch-based approaches. They assess the value of new samples for selecting individually and collectively as a batch, respectively.

Single sample acquisition takes the top b samples according to the score of the acquisition function to select a batch of size b. Several methods follow this selection scheme based on either epistemic uncertainty or representation score. For example, uncertainty-based methods try to select the most uncertain samples to acquire a batch. Many methods, such as EqualAL [45], Ensemble+AT [71], and CEAL [121], estimate uncertainty based on the output probabilities. Epistemic uncertainty, estimated using Entropy [105], is often

used as a strong baseline in several active learning works [45, 106, 99]. Some methods, namely BALD [53] and DBAL [40] employed a Bayesian approach using Monte Carlo Dropout [39] to measure the epistemic uncertainty. Representation-based methods aim to select the most representative samples of the dataset that are not yet covered by the labeled samples. Numerous adversarial learning-based methods utilize an auxiliary network to score samples based on this measure, including DAAL [122], VAAL [109], and WAAL [107]. For our study, we employ Entropy, EqualAL, and BALD to represent single-sample acquisition methods due to their direct applicability to segmentation tasks. We did not include deep ensemble-based methods due to their limited scalability and adversarial methods due to their hyperparameter sensitivity. In general, single-sample acquisition approaches select individually very informative samples but do not optimize the joint improvement obtained with the whole batch.

Batch-based acquisition methods acquire the whole batch of size b to maximize cumulative information gain. Sener *et al.* [103] formulated the acquisition function as a core-set selection approach based on the feature representations. It is a representation-based approach that selects the batch of samples jointly to represent the whole data distribution. BatchBALD [68] is a greedy algorithm that selects a batch of points by estimating the joint mutual information between the whole batch and the model parameters. This method was also proposed to remedy the mode collapse issue, where the acquisition function collapses into selecting only similar samples (see Section 3.3.3 for details). However, it is limited to simple image classification datasets like MNIST [31] since its computation complexity grows exponentially with the batch size. Some more recent batch-based methods include k-MEANS++ [145], GLISTER [65], ADS [61], but these methods only evaluate on image classification tasks. For the study, we selected the Coreset method [103] to represent batch-based methods due to its effectiveness, simplicity, and easy scalability to the segmentation task.

3.1.3 Active Learning for Semantic Segmentation

Many of the approaches mentioned above mainly focus on image classification. Lately, a few works have proposed to solve tasks involving higher annotation costs like object detection [136], pedestrian detection [136], human pose estimation [78], and segmentation [60, 109].

Along with the task of active learning for image classification, we also focus on semantic segmentation since creating segmentation masks is an expensive labeling task. This makes semantic segmentation one of the most relevant tasks for active learning. Suggestive Annotation [135], Cereals [83], and VAAL [109] are a few previous works that have shown the applicability of deep active learning for semantic segmentation.

When applied to semantic segmentation, active learning methods must choose which area of the image is to be considered for the acquisition: the full image [109], superpixels [14], polygons [86, 45], or each pixel [106]. There is no common understanding so far of which approach is cheaper and more effective. Thus, our study uses the straightforward image-wise selection and annotation procedure.

Most existing methods for segmentation are based on the model's uncertainty for the input image, where the average score over all pixels in the image is used to select top-k images. Entropy [105] (estimated uncertainty) is a widely used active learning baseline for selection. This function computes per-pixel entropy for the predicted output and uses the averaged entropy as the final score. EqualAL [45] determines the uncertainty based on the consistency of the prediction on the original image and its horizontally flipped version. The average value over all the pixels is used as the final score. BALD [53] is often used as a baseline in existing works. It is employed for segmentation by adding dropout layers in the decoder module of the segmentation model and then computing the pixel-wise mutual information using multiple forward passes. Coreset [103] is a batch-based approach that was initially proposed for image classification, but it can be easily modified for segmentation. For e.g., the pooled output of the ASPP [19] module in the DeepLabv3+ [21] model can be used as the feature representation for computing distance between the samples. Some other methods [109, 66, 107] use a GAN model to learn a combined feature space for labeled and unlabeled images and utilize the discriminator output to select the least represented images. Our study includes Entropy, EqualAL, BALD, and Coreset approaches for the analysis, along with the random sampling baseline. Most AL methods for semantic segmentation use single-sample acquisition and show superior performance over batch acquisition function Coreset. This chapter also studies the integration of these methods with semi-supervised learning.

3.1.4 Semi-supervised Active Learning

Active learning uses a pool of unlabeled samples only for selecting new samples for annotation. However, this pool can also be used for semi-supervised learning (SSL), where the objective is to learn jointly from labeled and unlabeled samples.

Most representation-based AL methods use unlabeled samples to learn the underlying distribution, but only a few methods use semi-supervised learning to improve their selection criteria [35, 36, 39, 41]. Sinha et al. [39] used an unlabeled pool to learn its distribution against the distribution of labeled samples. Still, they did not take advantage to improve the feature representation of the target model itself. Sener et al. [36] have also previously shown the advantage of using the unlabeled pool for learning the target model. Wang et

al. [41] also explored the usage of the most-certain samples from the un-labeled pool using pseudo-labeling, but the pseudo-labeling process can easily propagate erroneous labels if not tuned properly. Ravanbakhsh et al. [35] proposed a GAN-based approach to use the unlabelled pool and utilize the discriminator score to query low-confident samples for active learning. Recently, two concurrent open-source works [2, 3] have also shown some similar findings to our work. However, they are restricted to only image classification.

The combination of SSL and AL has been used successfully in many other contexts, such as speech understanding [20, 9] and pedestrian detection [30]. Some recent works have also studied active learning methods with the integration of SSL for segmentation, but their scope is limited only to special cases like subsampled driving datasets [29] or low labeling budget [27], both cases with only single-sample acquisition methods.

Although modern semi-supervised learning works have been shown to be very effective in using unlabeled samples in the dataset to reach near 100% supervised performance, it is an understudied topic in combination with active learning. Overall, a broader understanding of whether the usage of unlabeled samples is useful or detrimental is not clear. Our work provides an overview of the integration of SSL and active learning for the image classification and semantic segmentation task. We study this integration over datasets with different redundancy levels, under different labeling budgets, and with single-sample and batch-based methods. Our findings explain when this integration is effective and boosts the active learning method.

3.1.5 Current Benchmarks

Current AL methods for image classification are mostly tested on CIFAR-10 and CIFAR-100 datasets, which are perfectly balanced. Some recent works [66] have also tested on higher resolution datasets like Caltech101, which is naturally imbalanced. In this work, we also use CIFAR-10 and CIFAR-100 for our study on image classification. However, we evaluate the AL methods with various new settings like strong augmentation, integration of SSL, and low-annotation budget. Current AL methods for semantic segmentation are usually evaluated on driving datasets due to the industrial focus on autonomous driving. These datasets include Cityscapes [25], BDD100K [137] and CamVid [13]. Some works [85] evaluate more generic datasets like PASCAL-VOC [35]. Medical datasets [143, 23, 108] are also common for the AL studies due to extremely high annotation cost. In this work, we focus on driving datasets and introduce a more realistic driving AL task.

3.2 Active Learning for Image Classification

Model training techniques for deep neural networks are advancing very fast. Often this progress is ignored by deep active learning studies. In recent times, many interesting AL methods have been proposed; however, they treat the AL problem independently of the training process of the model. They often ignore the progress of different training elements like optimizers, data augmentation, and learning techniques.

In this section, we assess the performance of state-of-the-art AL methods for image classification and compare them with the integration of data regularization and a state-of-the-art semi-supervised learning approach. We also challenge the previously proposed methods under a low annotation budget where the initial model is trained with fewer labeled samples, followed by a few new sample selections at each AL cycle. We validate our experiments using at least one recent approach from each of the categories of AL methods as defined in the related work Section 3.1.

3.2.1 Integration of AL with Label-efficient Learning

Active Learning with Data Augmentation Recently, various regularization techniques have been proposed to improve model generalization with minimal labeled data. Although these methods show consistent success in various applications, they have been ignored by the works on active learning. In this work, we study the effect of one such regularization: *data augmentation*. Data augmentation is a widely accepted regularization technique, which increases the power of machine learning models, particularly when there is little labeled data. Nevertheless, several latest AL works [11, 109] resort to either not using any augmentation during training or only doing simplistic augmentations like horizontal flipping. The behavior of active learning under the influence of strong data augmentation is largely unknown. In the experiments, we apply strong augmentations, including color and geometric augmentations, during the training phase of the model. The acquisition function selects samples based on this model.

Integration of Active Learning with Semi-supervised Learning A largely common practice in the previous works has been to utilize the unlabeled pool only for sampling, although it is available throughout the learning process (otherwise, one could not sample from it) and could be used more rigorously. Using semi-supervised learning, we can utilize this unlabeled pool for training the model itself and thus learn an improved query function

using unlabeled samples. To this end, we employed the UDA [130] semi-supervised learning method. UDA applies a consistency loss between differently augmented unlabeled samples to learn from unlabeled samples. We integrated SSL into the AL methods by training the model using the UDA objective and defining the query function based on this model. In each cycle, the target model is trained using UDA instead of the standard supervised training.

Active Learning under Low-annotation Budget We observed in the literature that there is an inconsistency in the methods' behavior when switching from CIFAR-10 to CIFAR-100. This challenges the principal assumption of active learning that a dedicated selection strategy always improves over a random selection of samples. *We ask whether active learning benefits from a low-budget setting, where every sample is particularly crucial.* In certain applications, such as medical image analysis, already 10000 annotated samples can be very costly. Thus, training with only a few labeled samples in the beginning is attractive. We study the behavior of active learning methods where the initial and sampling annotations budget is 10 to 20 times smaller than usually studied in previous works.

3.2.2 Experiment Setup

We evaluate and compare following **baseline methods**:

- *Random*: A new set of samples is selected randomly from the unlabeled pool and is added to the labeled pool with annotations.
- *Entropy*: [105] is an information-theoretic measure used as an uncertainty metric for sampling. This method naively selects samples for which the pseudo-probabilities predicted by the softmax classifier have the highest entropy.

$$H(y|x_u) = -\sum_c (p(y=c|x_u)) \log(p(y=c|x_u)). \tag{3.2}$$

  For the entropy method, we use the softmax output of the final fully-connected layer to calculate the entropy of the prediction.

- *Ensemble with Variation Ratio (ENS-varR):* The second method, which selects samples based on an uncertainty criterion, relies on using ensembles. It has been shown to consistently outperform all other uncertainty-based approaches for active learning by Beluch *et al.* [11]. The core of the method is to calculate the variation ratio (varR)

metric given as the proportion of predicted class labels that are not the modal class prediction:

$$varR = 1 - \frac{f_m}{T}, \tag{3.3}$$

where f_m is the frequency of the modal class and T is the number of ensemble members. This heuristic is motivated by the query-by-committee algorithm proposed by Seung *et al.* [104]. The query function selects the samples with larger varR values. The ensemble is only used for sample querying - the target performance is still reported for a single model. Similar to Beluch *et al.*[11], we use an ensemble of 5 models for our experiments.

- *Core-set*: This type of method selects a batch of samples such that the performance of the model trained on the labeled set matches the performance of the model trained on the whole dataset [96]. The recent core-set approach proposed by Sener *et al.* [103] casts the core-set selection problem as a k-center problem and proposes a robust k-center approach. The proposed approach chooses a subset such that the largest distance between the chosen point and unlabeled points is minimized in the feature space. For the core-set approach, we make use of the k-center greedy implementation since it is much faster and only performs marginally worse than the robust version.

- *Learning Loss (LL)*: This method [136] proposes a loss prediction module that is attached to the target network to estimate the loss value of the unlabeled samples. The samples with the largest predicted loss are selected for annotation. This auxiliary module is trained to preserve the pairwise ranking of the original loss values, which is imposed using a hinge loss function over random pairs of samples in a minibatch.

- *Unsupervised Data Augmentation (UDA)*: UDA [130] is a semi-supervised learning method for image classification. It uses consistency regularization to learn from unlabeled samples along with AutoAugment [26] and other augmentation techniques to reduce overfitting. We selected this method because: 1) it shows state-of-the-art performance, 2) it is based on a simple idea and is easy to implement. Also, the method performs well even when the number of labeled samples is very small. Our implementation used online data augmentation instead of the offline one in the original work [130].

Datasets. We evaluate the methods on the CIFAR-10 and CIFAR-100 datasets. Both datasets contain the same set of 60,000 images, assigned to 10 and 100 classes, respectively.

The training and test set contains 50,000 and 10,000 images, respectively. CIFAR-10 is the most commonly tested dataset in the field of active learning. CIFAR-100 is an extension with 100 classes, which makes the task more challenging. The initial labeling budget is $\mathcal{B}_i = 5000$, and the sampling budget is $\mathcal{B}_s = 2500$ labels for each cycle. We tested this configuration for 6 sampling cycles (*i.e.* going from 10% to 40% labeled samples). In the first step, we randomly sampled a class-balanced subset of samples from the unlabeled pool.

Training details. For the network architecture, we consistently use the Wide-Resnet-Network [140] with depth=28 and width=2 (WRN-28-2). We select WRN due to its efficiency and widespread adoption. WRN-28-2 contains only 1.5M parameters showing close-to-state-of-the-art performance on CIFAR datasets. The WRN-28-2 classification network is optimized using an SGD optimizer with a base learning rate of 3e-2, momentum of 0.9, and weight decay rate of 5e-4. We use a cosine learning rate schedule for training each model. We trained all AL methods (without SSL methods) for 150 epochs per sampling cycle with a batch size of 64. We train the semi-supervised AL methods for 50k iterations per sampling cycle with a batch size of 64 for the labeled loss and a batch size of 320 for the unlabeled loss. We mask out unlabeled examples whose highest probabilities across categories are less than 0.6 and set the softmax-temperature scaling constant to 0.5. Other hyperparameters are used exactly as proposed in [130]. Our implementation is based on the open-source toolbox Pytorch [95].

All results are shown as performance curves. We report the mean performance over 3 trials with different initial labeled sets for all single model-based methods and over 2 trials for ensemble-based methods due to higher computation cost and lower variance.

LL method usually starts with a higher initial performance due to the extra regularization effect from the loss-prediction module. All other methods start from similar initial performance with a slight difference due to the model variance. This variance is more prominent in the beginning due to the overfitting effect on a small labeled set.

Evaluation metrics. We evaluate AL methods in different data budget settings, referred to as the $\mathcal{B}_i$-$\mathcal{B}_s$ setting, where $\mathcal{B}_i$ is the initial label budget, $\mathcal{B}_s$ is the sampling-label budget, and $\mathcal{B}_i$,$\mathcal{B}_s$ refer to the number of labeled images. Images are sampled randomly to fulfill the initial label budget. For the subsequent steps, images are sampled using the AL acquisition function with the sampling-label budget. We test these datasets with $5K - 2.5K$, $500 - 500$, and $250 - 250$ settings.

We use mean Intersection over Union (mIoU) to evaluate the performance of the model at each AL cycle step. For the evaluation of the active learning method, we use two metrics:

Area Under the Budget Curve (AUC@$\mathcal{B}$) and mean Intersection over Union at a budget $\mathcal{B}$ (mIoU@$\mathcal{B}$).

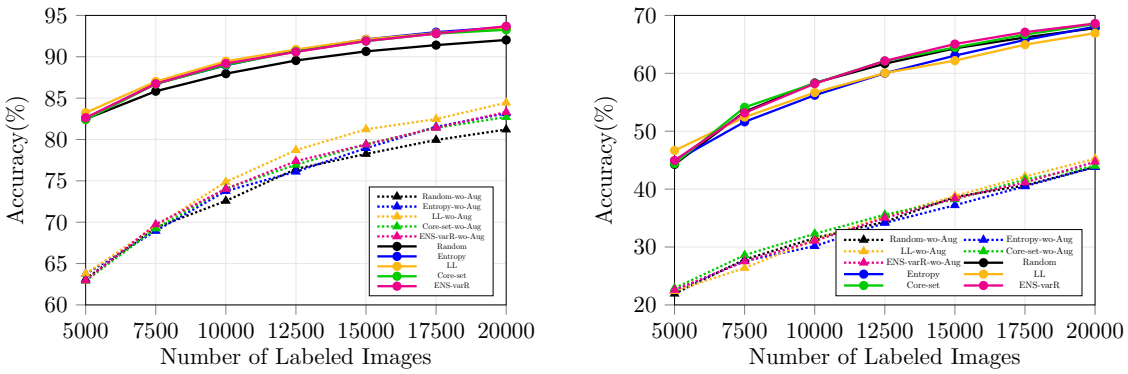
- **AUC@$\mathcal{B}$** is the area under the performance curves, shown in Figure 3.3 and 3.4. It captures a cumulative score of the AL performance curve up to a budget $\mathcal{B}$, where $\mathcal{B}$ is the number of labeled images. We use a total budget of $\mathcal{B}$=20K in the 5K-2.5K setting for CIFAR-10 and CIFAR-100 datasets. We use $\mathcal{B}$=2K for CIFAR-10 in 250-250 setting and $\mathcal{B}$=4K for CIFAR-100 in 500-500 setting. We use the following formula to compute the Area Under the Budget Curve(AUC@$\mathcal{B}$) at a total budget $\mathcal{B}$, where $\mathcal{B}$ is the percentage of the labeled dataset:

$$AUC@\mathcal{B} = \sum_{i=1}^{i=N} \frac{(b_{i+1} - b_i)(p_i + p_{i+1})}{2} \tag{3.4}$$

  ,where N is the number of AL acquisition steps, b_i is the percentage of the labeled dataset from the whole dataset at step i, and p_i is the performance of the model in mIoU(%) at step i.

- **Acc@$\mathcal{B}$** reports the Accuracy of the model after using a certain labeling budget $\mathcal{B}$. We report performance at an intermediate labeling budget to clearly see the ranking of the AL methods.

3.2.3 Results



(a) CIFAR-10 (b) CIFAR-100

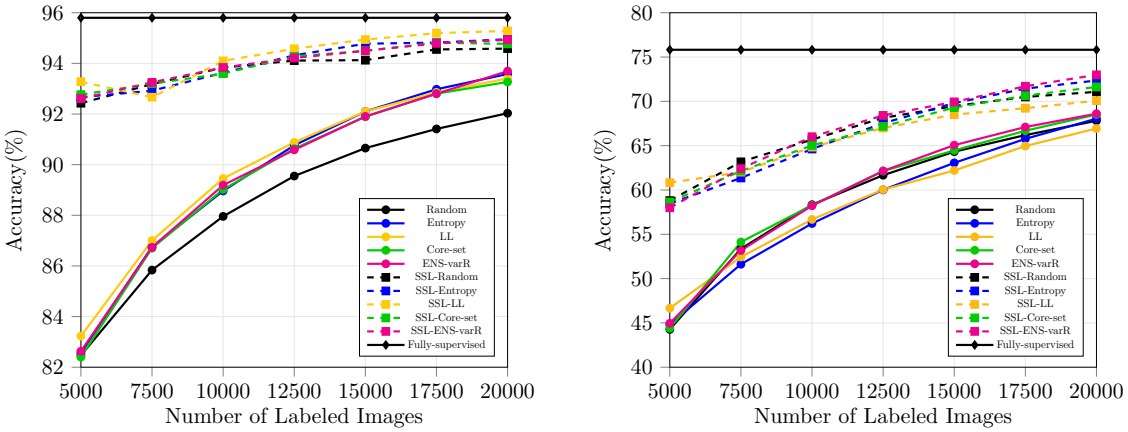
Fig. 3.3 Using data augmentation on CIFAR-10 significantly improves the performance of active learning methods and makes the relative difference between them less pronounced. The performance of AL methods on CIFAR-100 improves significantly when using up-to-date image augmentation. Results without augmentation are denoted as 'X-wo-Aug'.

AL Method **Metric** →	**Strong Aug.**	**CIFAR-10** **Acc**	**AUC**	**CIFAR-100** **Acc**	**AUC**
Random w/o Aug	✗	76.43	44.87	34.35	20.59
Entropy w/o Aug	✗	76.11	45.28	34.16	20.27
LL w/o Aug	✗	78.71	46.09	35.28	20.77
Coreset w/o Aug	✗	76.94	45.39	35.57	20.98
ENS-varR w/o Aug	✗	77.39	45.51	35.03	20.68
Random w/ Aug	✓	89.55	53.27	61.67	35.99
Entropy w/ Aug	✓	90.76	53.96	60.01	35.31
LL w/ Aug	✓	**90.88**	**54.06**	60.04	35.31
Coreset w/ Aug	✓	90.63	53.89	62.13	36.21
ENS-varR w/ Aug	✓	90.58	53.94	**62.15**	**36.25**
100%	✓	95.80	57.48	75.82	45.49

Table 3.2 Active Learning results on CIFAR-10 and CIFAR-100 datasets. AUC@20K and Acc@12.5K metrics are reported. The table compares AL results with and without the usage of strong data augmentation during the training of the model.

Integration with Data Augmentation In this experiment, we validated the importance of elaborate up-to-date image augmentation for the performance of AL methods. We first evaluated all methods without any augmentation. Subsequently, we evaluated the same methods with augmentation, which includes using the AutoAugment policies found by Cubuk *et al.* [26] , cutout [32], horizontal random flipping, and random cropping. Figure 3.3 shows that without using any augmentation, all AL methods clearly perform better than the random baseline. The LL method shows distinct improvement over other methods (matching the results from Yoo *et al.* [136]) and an overall improvement of 3.2% over the random baseline on the CIFAR-10 dataset. When the same experiment is performed with augmentation, all the methods improve drastically in absolute performance. However, the relative effect of using different AL methods becomes far less pronounced: all the AL methods show similar performance within a range of 0.4%. In conclusion, AL works well with data augmentation, but data augmentation blurs the differences between AL strategies: they all perform largely the same.

For completion, we further validate the importance of using up-to-date augmentation for AL methods on the CIFAR-100 dataset. We evaluate all methods with and without augmentation, similar to the CIFAR-10 experiment. The overall conclusion is also very similar: Without augmentation, the LL method shows a distinct improvement of 1.4% over the random baseline; with augmentation, all the methods improve by a large margin in absolute performance, but the relative difference between different methods becomes



(a) CIFAR-10 (b) CIFAR-100

Fig. 3.4 Combining AL methods with semi-supervised learning leads to significant performance improvement on CIFAR-10 compared to the raw AL case. Results shown in the large-budget setting with $\mathcal{B}_i = 5000, \mathcal{B}_s = 2500$. Integrating SSL and AL leads to overall performance improvement on CIFAR-100, however, not all combinations consistently outperform random sampling. Results shown in the large-budget setting with $\mathcal{B}_i = 5000, \mathcal{B}_s = 2500$.

insignificant and the relative ranking of different methods changes. Performance curves are shown in Figure 3.3b.

Integration with Semi-supervised Learning We refer to the integrated methods as SSL-X, where X is the name of the AL method. Figures 3.4a and 3.4b show a remarkably strong performance of the SSL method (SSL-Random) on CIFAR10 and CIFAR100: when using 5K random labeled samples, SSL almost reaches the same performance which AL methods achieved on 20K samples picked by the corresponding query functions. Also, for the remaining data ratios, there is a large performance gap between semi-supervised and active learning, both on CIFAR-10 and CIFAR-100. Clearly, semi-supervised learning makes much better use of the same data than active learning.

SSL and AL can be combined, which yields an improvement over raw SSL on CIFAR-10. The SSL-LL method performs best and shows an improvement over the random baseline by 0.7% after 6 cycles. However, on CIFAR-100, the relative ranking of the AL methods changes completely; SSL-LL performs worse than the other methods and struggles even to compete with the random selection method.

The same is true for raw active learning without SSL: on CIFAR-100, some active learning methods do not reach the performance of randomly drawing the samples to be labeled, shown in Figure 3.4b.

AL Method **Metric →**	**Strong Aug.**	**SSL**	**CIFAR-10** **mIoU**	**AUC**	**CIFAR-100** **mIoU**	**AUC**
Random w/ Aug	✓	✗	89.55	53.27	61.67	35.99
Entropy w/ Aug	✓	✗	90.76	53.96	60.01	35.31
LL w/ Aug	✓	✗	90.88	54.06	60.04	35.31
Coreset w/ Aug	✓	✗	90.63	53.89	62.13	36.21
ENS-varR w/ Aug	✓	✗	90.58	53.84	62.15	36.25
Random-SSL	✓	✓	94.11	56.33	68.14	40.19
Entropy-SSL	✓	✓	94.32	56.43	67.54	40.02
LL-SSL	✓	✓	**94.58**	**56.58**	66.99	39.70
Coreset-SSL	✓	✓	94.27	56.41	67.17	39.94
ENS-varR-SSL	✓	✓	94.21	56.44	**68.42**	**40.40**
100%	✓	✗	95.80	57.48	75.82	45.49

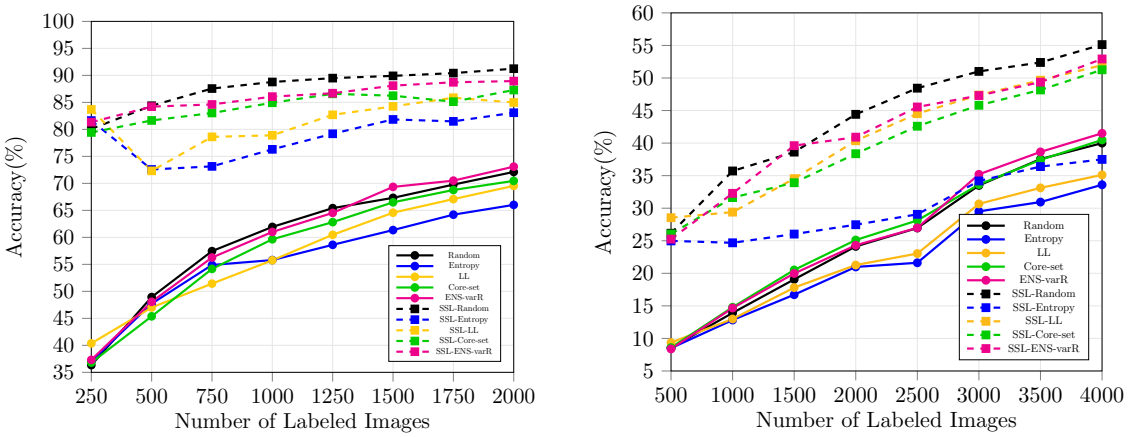
Table 3.3 Active Learning results on CIFAR-10 and CIFAR-100. AUC@20K and mIoU@12.5K metrics are reported. The table compares AL methods with and without semi-supervised learning.

High-budget vs Low-budget We explored such low-budget settings with $\mathcal{B}_i$ and $\mathcal{B}_s$ for each cycle set to 250 labels for CIFAR-10 and 500 labels for CIFAR-100. We tested this setting for 7 sampling cycles with a total budget of 2000 and 4000 labels for CIFAR-10 and CIFAR-100, respectively. We kept all the augmentation techniques from the previous experiments.

The results are shown in Figures 3.5a and 3.5b. None of the active learning methods consistently outperforms the random baseline, neither on CIFAR-10 nor on CIFAR-100. This always holds for the combination of active learning and semi-supervised learning, whereas for raw active learning, only ENS-varR could marginally outperform the random baseline. In fact, some techniques perform considerably worse than the random baseline, especially in conjunction with semi-supervised learning, showing that their selection strategy is counter-productive in the low-budget regime.

Comparison to Transfer Learning Oliver *et al.* [92] argued that transfer learning might be a preferable alternative to semi-supervised learning when a suitable labeled dataset is available for transfer learning. Following the recommendation, we compare the performance of the SSL-Random baseline with a fine-tuned ImageNet pre-trained network on CIFAR-10.

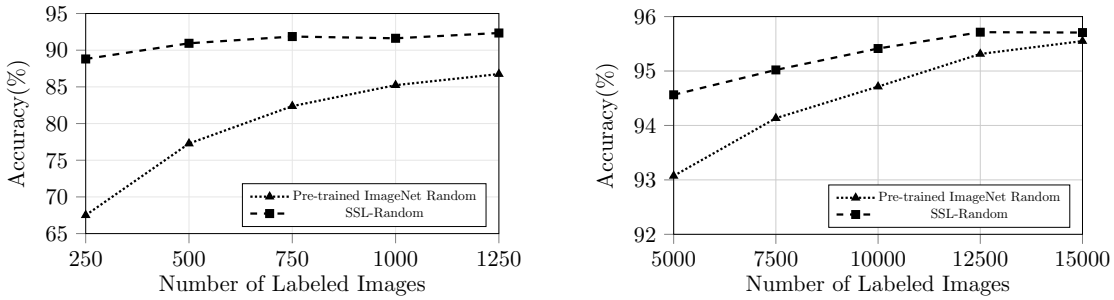
The ImageNet pre-trained network is fine-tuned only on the labeled samples. The experiment was conducted with Resnet-18 due to the availability of pre-trained ImageNet weights. We observe that the SSL-AL method clearly outperforms fine-tuning of a pre-trained ImageNet network in both high- and low-budget settings. We tested both budget setting for 4



(a) CIFAR-10 (b) CIFAR-100

Fig. 3.5 When evaluated in the low-budget regime ($\mathcal{B}_i = \mathcal{B}_s = 250$) on CIFAR-10, integrated SSL-AL methods are still better than their raw counterparts, however, SSL with random sampling shows the best performance. When evaluated in the low-budget regime ($\mathcal{B}_i = \mathcal{B}_s = 500$) on CIFAR-100, most integrated SSL-AL methods are still better than their raw counterparts but nothing beats SSL with random sampling.

sampling cycles. The corresponding results are shown in Figure 3.6a and 3.6b, respectively. This experiment shows that including an up-to-date semi-supervised learning algorithm in an active learning pipeline makes sense even when large pre-training data is available.



(a) Low-annotation budget (b) High-annotation budget

Fig. 3.6 The SSL-Random baseline clearly outperforms a fine-tuned network pre-trained on ImageNet in the low-budget setting. Results shown on CIFAR-10. The SSL-Random baseline clearly outperforms a fine-tuned network pre-trained on ImageNet in the large-budget setting. Results shown on CIFAR-10.

3.2.4 Conclusion and Proposed Evaluation Protocol

AL Method **Metric →**	**Strong Aug.**	**SSL**	**CIFAR-10** **mIoU**	**AUC**	**CIFAR-100** **mIoU**	**AUC**
Random w/ Aug	✓	✗	61.93	1.91	24.13	1.68
Entropy w/ Aug	✓	✗	55.79	1.76	20.98	1.43
LL w/ Aug	✓	✗	55.75	1.79	21.27	1.50
Coreset w/ Aug	✓	✗	59.64	1.85	25.11	1.72
ENS-varR w/ Aug	✓	✗	61.00	1.91	24.31	1.73
Random-SSL	✓	✓	**88.78**	**2.67**	**44.42**	**2.80**
Entropy-SSL	✓	✓	76.30	2.35	27.46	1.84
LL-SSL	✓	✓	78.90	2.44	40.36	2.57
Coreset-SSL	✓	✓	84.93	2.55	38.37	2.50
ENS-varR-SSL	✓	✓	86.07	2.60	40.91	2.65

Table 3.4 Active Learning results on CIFAR-10 and CIFAR-100. AUC@2K and mIoU@1K metrics are reported. The table compares AL methods with and without semi-supervised learning in a low-annotation budget setting.

In this work, we only studied active learning models under the influence of data augmentation, with the integration of semi-supervised learning under different annotation budgets.

Our experiments provide strong evidence that the current evaluation protocol used in active learning for image classification is sub-optimal, leading to wrong conclusions about the methods' performance and the state of the field in general.

Evaluating CIFAR-100, which is marginally different from CIFAR-10, dramatically changes the ranking of the methods. Applying state-of-the-art data augmentation significantly increases the scores of all methods, making them virtually indistinguishable in terms of the final performance. Modern semi-supervised learning algorithms applied in the conventional active learning setting show a higher relative performance increase than any of the active learning methods proposed in recent years. State-of-the-art active learning approaches often fail to outperform simple random sampling, especially when the labeling budget is small - a setting crucial for many real-world applications.

A recent work [91] has also explored the usage of model regularization like Stochastic weighted averaging [59] and Shake-shake(SS) [42] with active learning methods. Their observations confirm and provide conclusions very similar to our work.

Based on our observations, we formulate a more appropriate evaluation protocol and recommend using it for benchmarking future active learning methods for image classification.

1. AL methods should be evaluated on a broader range of datasets to assess their general robustness.

2. Evaluating AL methods with up-to-date network architectures and up-to-date augmentation techniques is vital.
3. There should always be a direct comparison between AL methods and SSL methods.
4. With the existing large-budget regime, AL methods should also be evaluated in the low-budget regime.

3.3 Active Learning for Semantic Segmentation

In this section, we assess the performance of state-of-the-art AL methods for semantic segmentation and compare them with the integration of semi-supervised learning. All methods are tested on datasets with different levels of redundancy and various levels of annotation budgets to understand how these methods behave under such diverse conditions which spam across different applications.

First, we provide some conceptual considerations which are important to understand the results later.

3.3.1 Conceptual Considerations

Our experiments show that the presence of redundant samples in the data distribution influences the choice of the acquisition function and the training regime that achieve the best performance. The main cause for this is the mode collapse issue, where the acquisition function collapses into selecting only similar samples. Here, we first discuss why and when this mode collapse occurs and how to remedy this issue. Then, we discuss ideal conditions for the successful integration of semi-supervised learning with active learning acquisition functions.

Redundancy can cause mode collapse Mode collapse in active learning refers to the circumstance that acquisition functions tend to select similar (redundant) samples when acquiring batches of data [68]. Since the selected similar samples contain highly redundant information, their annotation does not add much new value to the model performance. Figure 3.7 illustrates this mode collapse issue for a driving dataset case, where samples are selected from dense local feature space clusters using an epistemic uncertainty-based acquisition function. The mode collapse occurs when the dataset contains redundant samples, and the acquisition function is designed to select single samples based on some independent

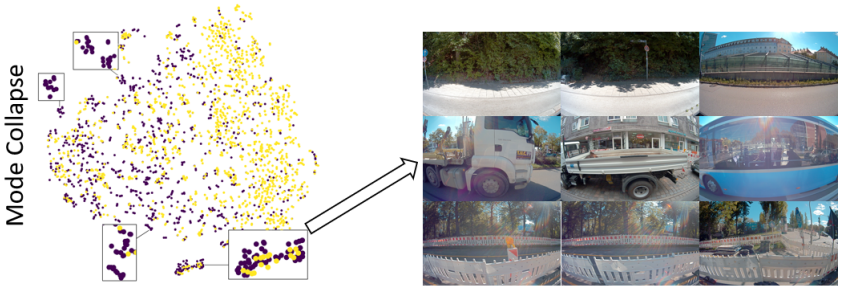


Fig. 3.7 T-SNE representations of the feature space. Yellow points represent the unlabeled data, and violet points represent the acquired data. The acquisition shows clear clusters in the feature space. The samples in the clusters correspond to similar images as depicted on the right. The *mode collapse* was observed while scoring samples independently based on uncertainty.

sample scores. Here, redundant samples tend to get similar scores from the acquisition function due to their similarity, i. e., their large overlap in information. Therefore, if one of those samples is selected due to a high score, other similar samples are also selected. This mode collapse effect occurs especially in Deep Active Learning scenarios since the acquisition of big batches is necessary to reduce the overall number of active learning cycles.

Existing deep active learning methods for semantic segmentation show that epistemic uncertainty is a good heuristic to select samples for annotation in common benchmarks. These strategies utilize single-sample acquisition functions and select the set of most valuable samples from the unlabeled pool based on the sampling budget. Since such methods were only tested on diverse datasets which are already curated for diversity, the mode collapse problem does not have a strong effect on their evaluation. However, this is not the case for many real-world applications. Redundancy occurs when there are repeated recordings of similar scenes or when the data is collected in a video format, like driving scenarios. A good acquisition function for such a redundant dataset must be aware of the batch's diversity to address the mode collapse issue. Intuitively, clustering-type approaches are ideal in redundant datasets since they select one sample from each local cluster avoiding single-sample selection traps like the mode collapse issue.

In this work, we argue that mode collapse is a common issue in real-world datasets and is largely ignored due to poor active learning benchmarks, which only cover diverse datasets. We probe previous AL methods for semantic segmentation over different diverse and redundant datasets. We design various redundant datasets based on the driving video dataset A2D2 [43] to reveal how the behavior of active learning methods changes with the level of redundancy in the dataset.

Requirements for Integration of Semi-supervised Learning and Active Learning Active learning methods use a pool of unlabeled samples only for selecting new samples for annotation. However, this pool can also be used by semi-supervised learning, where the objective is to learn jointly from labeled and unlabeled samples. In this work, we integrate semi-supervised learning with active learning in the context of semantic segmentation, an idea that was previously proposed for classification [103, 41, 86, 91]. In particular, we train the model using a semi-supervised learning objective, which impacts the resulting model and hence the acquisition function.

Successful integration can also be conceptually explained based on the underlying assumption of semi-supervised learning and the selection principle of the active learning approach. According to the *clustering assumption* of SSL, if two points belong to the same cluster, then their outputs are likely to be close and can be connected by a short curve [16]. In this regard, when labeled samples align with the clusters of unlabeled samples, the cluster assumption of SSL is satisfied, resulting in a good performance. Consequently, to maximize semi-supervised learning performance, newly selected samples must cover the unlabeled clusters that are not already covered by labeled samples. Only acquisition functions that foster this coverage requirement have the potential to leverage the additional benefits that arise from the integration of semi-supervised learning. A batch-based method, e.g., Coreset, selects samples for annotations to minimize the distance to the farthest neighbor. By transitivity, such labeled samples would have a higher tendency to propagate the knowledge to neighboring unlabeled samples in the cluster and utilize the knowledge of unlabeled samples using a semi-supervised learning objective and help boost the model performance. Similar behavior can also be attained using other clustering approaches that optimize for coverage.

3.3.2 Experiment Setup

Tested Approaches In our study, we test five active learning acquisition functions, including Random, Entropy, EqualAL, BALD, and Coreset. Here Entropy, EqualAL, and BALD approach represent single-sample, and Coreset represents the batch-based approach. All methods select the whole image for annotation. These methods are further described below, along with the segmentation-specific changes.

- *Random:* The samples are selected randomly for annotation from the unlabeled pool.

- *Entropy* [105]: This acquisition function uses per-pixel entropy as an estimation of the epistemic uncertainty for the predicted output. The final score for selection is the average entropy over all pixels. This method selects all top-scoring images.

- *EqualAL* [45]: The EqualAL approach determines the uncertainty based on the self-consistency between the prediction on the original image and its horizontally flipped version. The average uncertainty value over all the pixels is used as the final score. We use the EqualAL implementation, which trains using only cross-entropy loss to keep the baselines comparable.

- *BALD:* [53] The BALD approach is based on a Monte Carlo Dropout network to compute the pixel-wise Mutual Information of the classification. In our implementation, we employ dropout layers with a dropout ratio of 10% in the decoder layer and, during inference, compute 10 passes.

- *Coreset:* [103] The Coreset approach selects a batch of samples that cover the whole data distribution. It formulates this batch selection as a robust k-center selection problem. Coreset implements a greedy algorithm that iteratively selects unlabeled samples with maximum distance to the nearest neighbor of the so far selected samples. We utilize the k-center greedy approach since it is much faster and only performs slightly worse than the robust formulation. We use the ASPP module output in the DeepLabv3+ [21] model as the feature representation.

- *MCD setting:* Since the BALD method requires the introduction of Dropout layers into the architecture, we segregate the methods into two categories: With Monte Carlo Dropout (MCD) and without Monte Carlo Dropout layers. Random, Entropy, EqualAL, and Coreset are without MCD. BALD and Coreset-MCD are based on MCD. We compare methods in each category separately due to different architectures. We show fully-supervised performance, referred to as '100%' in the result tables, both with (100% MCD) and without MCD (100%) architectures.

Semi-supervised Learning To leverage the unlabeled samples, we use the semi-supervised learning s4GAN method [85]. It uses adversarial training to align the labeled and unlabeled data distribution and further uses self-training based on the GAN discriminator score. We pair all the used active learning approaches with SSL using this approach. This is marked by the suffix '-SSL' in the experiments. In particular, we train the model using an SSL objective, which impacts the resulting model and hence the acquisition function.

A	AL Method Metric →	SSL	Cityscapes mIoU	 AUC	A2D2 Pool-0f mIoU	 AUC
S	Random	✗	58.90	23.29	48.48	19.20
S	Entropy	✗	61.83	24.25	52.40	**20.37**
S	EqualAL	✗	62.41	24.32	**52.50**	20.35
B	Coreset	✗	60.89	23.89	51.14	19.88
S	Random-SSL	✓	60.72	23.85	49.69	19.60
S	Entropy-SSL	✓	60.61	23.93	50.80	19.90
S	EqualAL-SSL	✓	60.26	23.96	51.08	20.02
B	Coreset-SSL	✓	**63.14**	**24.47**	51.49	20.02
-	100%	✗	68.42	27.37	56.87	22.75
With MC-Dropout decoder						
S	BALD	✗	61.87	24.28	**52.82**	**20.32**
S	BALD-SSL	✓	61.13	23.89	52.29	20.14
B	Coreset-MCD	✗	60.60	23.78	49.99	19.45
B	Coreset-MCD-SSL	✓	**62.24**	**24.37**	51.76	19.97
-	100%-MCD	✗	67.07	26.83	56.47	22.59

Table 3.5 Active Learning results on Cityscapes and A2D2 Pool-0f. AUC@50 and mIoU@30 metrics are reported. A denotes the Acquisition method type. S and B denote the single-sample and batch-based acquisition, respectively.

Datasets Active learning methods are often evaluated on PASCAL-VOC and Cityscapes datasets, where PASCAL-VOC is naturally diverse while Cityscapes is diversified by sub-sampling from videos. In this work, we test on an additional driving dataset, A2D2, which is highly redundant. We evaluate the methods on these three datasets. To understand the nature of active learning methods over varying levels of redundancy in the dataset, we curate 5 smaller dataset pools from the large, original A2D2 dataset, described further below as A2D2-Pools.

- **Cityscapes** [25] is a driving dataset used to benchmark semantic segmentation tasks. The dataset was originally collected as videos from 27 cities, where a diverse set of images were selected for annotation. Due to the selection, Cityscapes cannot cover the redundant data scenario in our evaluation, although it was derived from videos. As we will see in the results, the nature of the active learning method changes when considering the raw form of data in a driving scenario, and pre-filtering, as done in Cityscapes, is sub-optimal compared to directly applying active learning on the raw data (see Section 3.3.4).

A	AL Method Metric →	SSL	PASCAL: 5-5 mIoU	 AUC	PASCAL: 10-10 mIoU	 AUC
S	Random	✗	70.70	13.92	72.13	28.85
S	Entropy	✗	70.38	13.94	73.72	29.17
S	EqualAL	✗	69.14	13.82	73.40	29.03
B	Coreset	✗	70.85	13.96	73.63	29.06
S	Random-SSL	✓	72.57	14.36	75.33	29.87
S	Entropy-SSL	✓	73.36	14.51	**76.08**	30.01
S	EqualAL-SSL	✓	**73.39**	**14.55**	75.89	**30.06**
B	Coreset-SSL	✓	72.88	14.46	75.91	30.03
-	100%	✗	77.00	15.40	77.00	30.80

Table 3.6 Active Learning results on PASCAL-VOC dataset in 5-5 and 10-10 settings. AUC@50 and mIoU@30 metric are reported. S and B denotes the single-sample and batch-based acquisition, respectively.

- **PASCAL-VOC** [35] is another widely used segmentation dataset. We use the extended dataset [48], which consists of 10582 training and 1449 validation images. It contains a wide spectrum of natural images with mixed categories like vehicles, animals, furniture, etc. It is the most diverse dataset in this study.

- **A2D2** [43] is a large-scale driving dataset consisting of 41277 annotated images with a resolution of 1920×1208 from 23 sequences. It covers an urban setting from highways, country roads, and three cities. It contains labels for 38 categories. We map them to the 19 classes of Cityscapes for our experiments. A2D2 provides annotations for every $\sim 10^{th}$ frame in the sequence and contains a lot of overlapping information between frames. Some consecutive frames are shown in Figure 3.9. We utilize 40135 frames from 22 sequences for creating our training sets and one sequence consisting of 1142 images for validation. The validation sequence '*20180925_112730*' is selected based on the maximum class balance. A2D2 represents the most redundant raw dataset in our study.

- **A2D2 Pools.** To obtain a more continuous spectrum between diverse and redundant datasets, we created five smaller dataset pools by subsampling the large A2D2 datasets. Each pool comprises 2640 images, which is comparable in size to the Cityscapes training set. Four pools are curated by subsampling the original dataset, while the fifth pool is created by augmentation. The first four pools, denoted by Pool-Xf (where X is 0, 5, 11, and 21), were created by randomly selecting samples and X consecutive frames for each randomly selected sample from the original A2D2 dataset. Pool-0f

A	AL Method Metric →	SSL	Pool-5f mIoU	 AUC	Pool-11f mIoU	 AUC	Pool-21f mIoU	 AUC	Pool-Aug mIoU	 AUC
S	Random	✗	47.58	18.69	44.61	17.76	44.52	17.67	43.80	17.15
S	Entropy	✗	49.96	19.48	47.43	18.52	46.08	18.21	44.51	17.33
S	EqualAL	✗	49.50	19.29	47.14	18.44	46.32	18.18	44.24	17.29
B	Coreset	✗	50.08	19.44	47.72	18.69	46.68	18.38	44.70	17.54
S	Random-SSL	✓	47.92	19.03	45.25	18.02	46.27	18.19	44.17	17.29
S	Entropy-SSL	✓	48.78	19.31	47.53	18.56	46.93	18.43	44.50	17.47
S	EqualAL-SSL	✓	48.80	19.28	46.50	18.39	47.11	18.54	44.81	17.56
B	Coreset-SSL	✓	**50.44**	**19.69**	**48.99**	**19.01**	**47.62**	**18.69**	**45.81**	**17.74**
-	100%	✗	53.25	21.30	48.85	19.54	49.23	19.69	46.03	18.41
With MC-Dropout decoder										
S	BALD	✗	**50.40**	19.29	47.85	18.74	46.78	18.57	**45.53**	**17.80**
S	BALD-SSL	✓	50.33	19.62	47.34	18.61	47.06	18.57	45.16	17.72
B	Coreset-MCD	✗	**50.40**	19.49	47.67	18.61	46.86	18.35	44.74	17.50
B	Coreset-MCD-SSL	✓	50.28	**19.65**	**48.60**	**18.96**	**47.73**	**18.75**	45.37	17.75
-	100%-MCD	✗	53.82	21.53	50.86	20.34	50.43	20.17	46.62	18.65

Table 3.7 Active Learning results on A2D2-Pool5f, A2D2-Pool11f, A2D2-Pool-21f, and A2D2-PoolAug. AUC@50 and mIoU@30 metrics are reported. S and B denotes the single-sample and batch-based acquisition, respectively.

contains only randomly selected images. We assume that the consecutive frames contain highly redundant information. Therefore, the pool with more consecutive frames has higher redundancy and lower diversity. The fifth pool, Pool-Aug, contains augmented duplicates in place of the consecutive frames. We create five duplicates of each randomly selected frame by randomly cropping 85% of the image area and adding color augmentation (see Figure 3.8).

Which dataset is diverse or redundant? We would like to clarify how we tag a dataset as diverse or redundant. Extreme cases like PASCAL-VOC can be easily tagged as diverse, and A2D2 original and A2D2-Pool-5f/11f/21f can be tagged as redundant. However, it is hard to put a redundant/diverse tag for many datasets in the middle of the spectrum. Cityscapes and A2D2-Pool-0f fall in this spectrum since they are curated by sparsely selecting from large video stream data. We consider them as non-redundant/diverse for our study since they behave more like diverse datasets.

Datasets visualization Figure 3.9 shows examples of the A2D2 and the Cityscapes dataset. Each row shows three temporally consecutive frames in both labeled datasets. We clearly observe that the images in the A2D2 dataset have high-overlapping information, whereas

Fig. 3.8 A2D2 Pool-Aug. Left: the original image. Right: the duplication through color augmentation and random cropping of the original image

images in the Cityscapes dataset are quite diverse. Therefore, to create our redundancy experiments, we chose the A2D2 dataset as the base dataset.

Training details We used the DeepLabv3+ [21] architecture with Wide-ResNet38 (WRN-38) [128] backbone for all our experiments. The backbone WRN-38 is pre-trained using ImageNet [30]. For the supervised learning setting, the model is trained using the SGD optimizer with a base-learning rate of $1e-3$, momentum of 0.9, and a weight decay of $5e-4$. We utilize a polynomial learning rate scheduler with a batch size of 8 and train a model in each AL cycle for 100 epochs. The model is trained with data augmentations, including random cropping and random horizontal flipping. Input image size is 256×512 for Cityscapes and A2D2 datasets and 321×321 for the PASCAL-VOC dataset.
We utilize the s4GAN [85] method for semi-supervised learning (SSL). We use the same training setting for the segmentation model as in the supervised learning setting. We use the same hyperparameters as mentioned in [85], except for the learning rate of the discriminator, which is set to $2.5e-5$ for Cityscapes and A2D2 experiments. We add 3 dropout layers with a dropout rate of 0.1 in the decoder of the segmentation model for all the MCD-based AL methods.

Evaluation metrics We use mean Intersection over Union (mIoU) to evaluate the performance of the model at each AL cycle step. For the evaluation of the active learning method, we use two metrics: Area Under the Budget Curve (AUC@$\mathcal{B}$) and mean Intersection over Union at a budget $\mathcal{B}$ (mIoU@$\mathcal{B}$). **AUC@$\mathcal{B}$** is the area under the performance curves, shown in Figure 3.10 and 3.11. It captures a cumulative score of the AL performance curve up to a budget $\mathcal{B}$, *where $\mathcal{B}$ is the percentage of the labeled dataset size.* For the experiments on A2D2 pools, we use a total budget of $\mathcal{B}$=50 in the 10-10 setting. For PASCAL-VOC, we run three experiments with $\mathcal{B}$=10, 25, and 50 in 2-2, 5-5, and 10-10 settings, respectively. For Cityscapes, we experiment with B=50 in the 10-10 setting. **mIoU@$\mathcal{B}$** reports the perfor-

Fig. 3.9 Consecutive images from the Cityscapes and A2D2 datasets. This shows even the consecutive images in the Cityscapes dataset are different and diverse, whereas consecutive frames in the A2D2 dataset are very similar, containing redundant information.

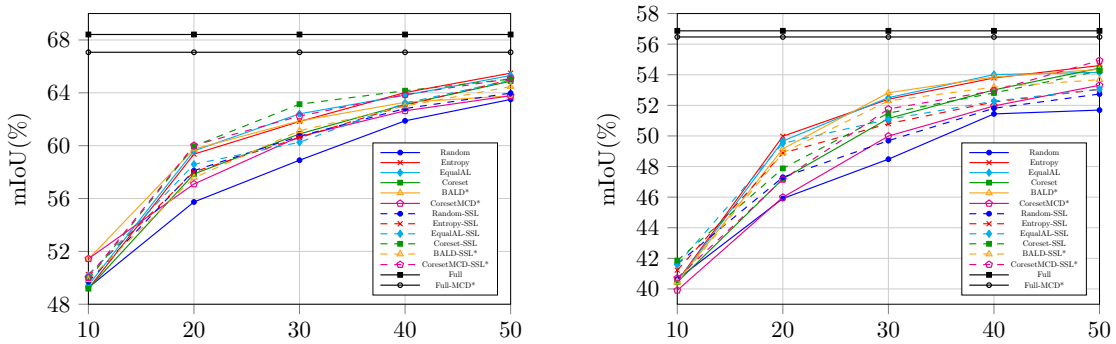
mance of the model after using a certain labeling budget $\mathcal{B}$. We report performance at an intermediate labeling budget to clearly see the ranking of the AL methods.

3.3.3 Results

Here, we answer the three questions raised in the introduction of the chapter concerning the behavior of active learning methods w.r.t data distribution in terms of redundancy, integration of semi-supervised learning, and different labeling budgets. For each experiment, we compare random sampling, single-sample, or batch-based acquisition approaches.

Impact of Dataset Redundancy

Table 3.5 and Figure 3.10 show the results on Cityscapes and A2D2 Pool-0f. For both datasets, the single-sample (S) method, EqualAL, performs the best in the supervised-only setting. Table 3.6 and Figure 3.12 shows the results obtained on the PASCAL-VOC dataset in 5-5 and 10-10 settings. Single-sample-based methods perform the best in the 10-10



(a) Cityscapes (b) A2D2:Pool-0f

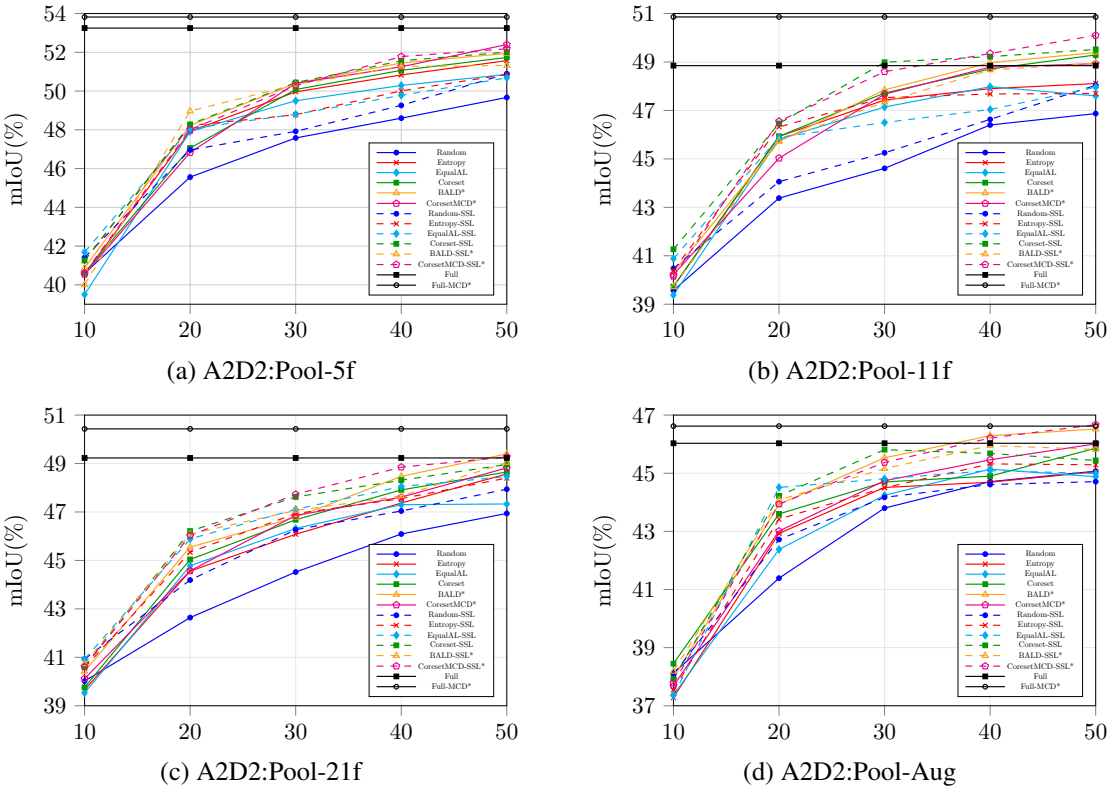
Fig. 3.10 Results on diverse driving datasets. Active learning performance curves on Cityscapes and A2D2:Pool-0f. X-axis shows the percentage of labeled dataset. The methods which utilize MC-dropout in their network architecture are marked with $*$, and are only comparable to other methods with MC-dropout.

setting, whereas Coreset performs the best in the 5-5 AL setting by a marginal gap w.r.t. random baseline. Table 3.7 and Figure 3.11 show the results for the redundant datasets. The batch-based Coreset method consistently performs the best in all four datasets in the supervised-only setting.

Diverse datasets need a single-sample method, and redundant datasets need a batch-based method. We observe that the order of best-performing models changes based on the level of redundancy in the dataset. Single-sample-based acquisition functions perform best on diverse datasets, whereas batch-based acquisition functions perform best on redundant datasets. We attribute this reversed effect to the mode collapse problem, where, for redundant datasets, single-sample acquisition methods select local clusters of similar samples. Diverse datasets are devoid of this issue as they do not possess local clusters due to high diversity across samples. Therefore, diversity-driven acquisition is not critical for diverse datasets.

This observation is consistent for PASCAL-VOC, where single-sample-based uncertainty-type methods perform better than batch-based and random methods in the high-budget setting. The difference between the methods is only marginal here since most acquired samples add ample new information due to the highly diverse nature of the dataset. This difference further diminishes w.r.t. random baseline with a lower labeling budget (*e.g.* 5-5) since any learned useful bias also becomes weaker. The observations for the 5-5 setting tend towards a very low-budget setting which is further analyzed later in this section.

Mode collapse analysis. Here, we analyze and visualize the above-mentioned model collapse issue. We provide a qualitative analysis of the mode collapse issue on the redundant A2D2 Pool-21f. We plot the feature representations using t-SNE to show the selection



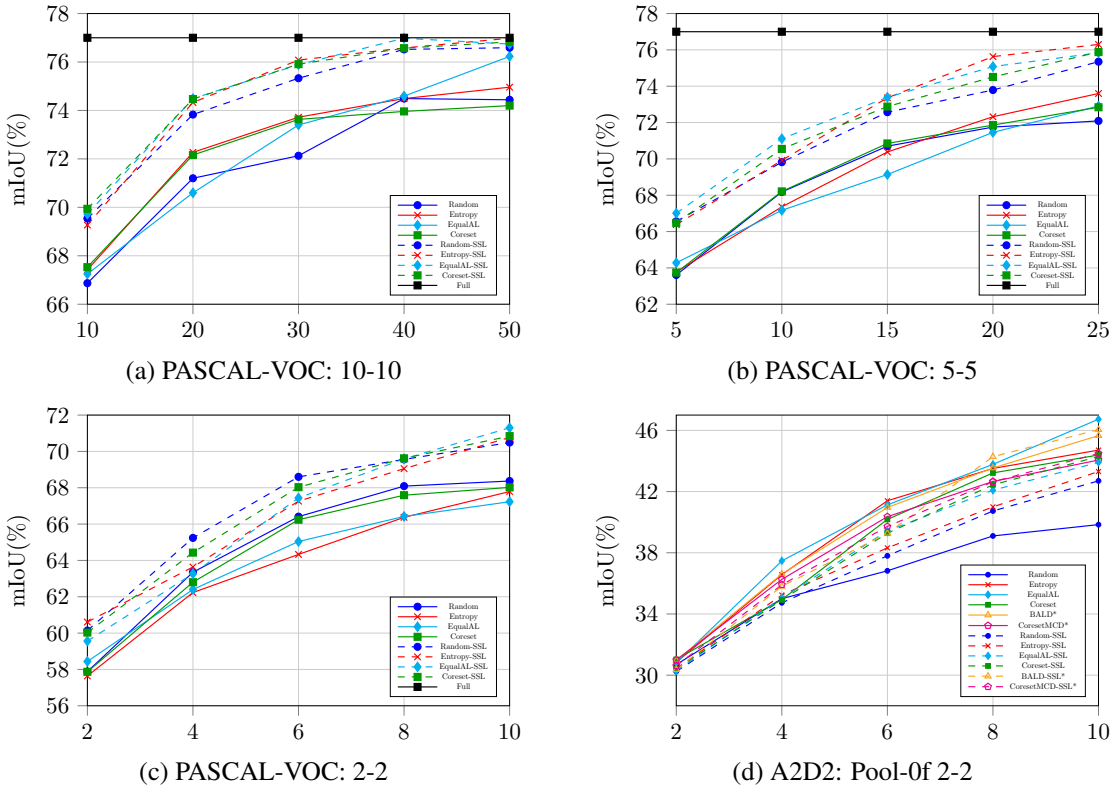
(a) A2D2:Pool-5f (b) A2D2:Pool-11f

(c) A2D2:Pool-21f (d) A2D2:Pool-Aug

Fig. 3.11 Results on redundant datasets. Active Learning performance curves on A2D2 dataset: Pool-5f, Pool-11f, Pool-21f, and Pool-Aug. The X-axis shows the percentage of labeled datasets. The methods which utilize MC-Dropout in their network architecture are marked with $*$, and are only comparable to other methods with MC-Dropout.

process for a single-sample-based Entropy function and batch-based Coreset function, shown in Figure 3.13. It shows that Entropy acquisition selects many samples within local clusters, which are similar samples with overlapping information. This yields a suboptimal use of the annotation budget. In contrast, Coreset acquisition has a good selection coverage and avoids this mode collapse.

In this work, we argue that mode collapse is a common issue in many real-world datasets, containing similar samples. A good acquisition function for such datasets must be aware of the batch's diversity to address the mode collapse issue. It is largely ignored due to the narrow scope of existing AL benchmarks like PASCAL-VOC and Cityscapes, which only cover diverse datasets.



(a) PASCAL-VOC: 10-10 (b) PASCAL-VOC: 5-5

(c) PASCAL-VOC: 2-2 (d) A2D2: Pool-0f 2-2

Fig. 3.12 Active learning performance curves on PASCAL-VOC and A2D2:Pool-0f. X-axis shows the percentage of labeled dataset. The methods which utilize MC-Dropout in their network architecture are marked with $*$, and are only comparable to other methods with MC-Dropout.

Systematic Integration of SSL

For all redundant datasets, the Coreset-SSL approach consistently performs the best; see results in Table 3.7 and Figure 3.7. For diverse datasets, SSL integration is also helpful, but there is no consistent best approach. For the PASCAL-VOC dataset, single-sample-based methods with SSL show the best performance, shown in Table 3.6. For Cityscapes, Coreset-SSL outperforms all other approaches; see Table 3.5 and Figure 3.10. For A2D2-Pool0f, Coreset-SSL improves over Coreset, but the single-sample acquisition method BALD approach shows the best performance.

Redundant datasets favor the integration of batch-based active learning and semi-supervised learning. The batch-based acquisition function Coreset always profits from the integration of SSL. Coreset aligns well with the SSL objective since Coreset selects samples from each local cluster, thus covering the whole data distribution. This assists SSL in obtaining maximum information from the unlabeled samples, as discussed in Section 3.3.1. This effect is especially strong in the redundant A2D2 pools, where Coreset-SSL always

improves over Coreset and also shows the best performance. In contrast, SSL integration for single-sample methods is either harmful or ineffective, except for the PASCAL-VOC dataset. Interestingly, in Pool-11f, some Coreset-SSL methods even outperform the 100% baseline with less than 30% labeled data. This indicates that some labeled redundant samples can even harm the model (see Figure 3.11), possibly due to data imbalance. For Cityscapes, SSL with Coreset yields significant improvement, and SSL even changes the ranking of the methods. We see that EqualAL performs the best in the supervised-only setting, whereas Coreset-SSL surpasses all methods. This slight anomaly in the case of Cityscapes happens because the advantage due to the combination of SSL and batch-based method is greater than the advantage of using single-sample methods in non-redundant datasets. For diverse PASCAL-VOC, all methods align well with SSL. All methods perform well with no clear winner method since all selection criteria select samples that provide good coverage of the data distribution.

Low Annotation Budget

Active learning is volatile with a low budget. Experimenting with PASCAL-VOC in the 2-2 budget setting, Random-SSL performs the best, i.e., semi-supervised learning without an active learning component (see Table 3.8 and Figure 3.12). We believe that active learning fails in this setting because it fails to capture any helpful bias for selection in such a low-data regime with diverse samples. Our observations in this low-budget setting confirm and provide stronger empirical support for similar behavior observed in [86]. For A2D2 Pool-0f and Cityscapes in the 2-2 setting (see Table 3.9 and 3.8), the single-sample acquisition performs the best, while its SSL integration is detrimental. These methods possibly learn some useful bias due to the specialized driving domain. For redundant datasets in low-budget settings, the batch-based acquisition is still the most effective way. However, SSL does not contribute any additional improvements due to insufficient labeled samples to support learning from unlabeled samples. Overall, we observe a highly volatile nature of active learning in conjunction with a low budget. The ideal policy transitions from random selection towards batch-based acquisition, as the dataset redundancy goes from low to high.

3.3.4 An exemplar case study: A2D2-3K task

Previous active learning works on semantic segmentation cover only the combination of a diverse dataset and a high annotation budget. In contrast, the collected raw data can be quite redundant, like in video datasets. To study this missing redundant setting, we propose a new active learning task A2D2-3K for segmentation based on the A2D2 dataset. The aim

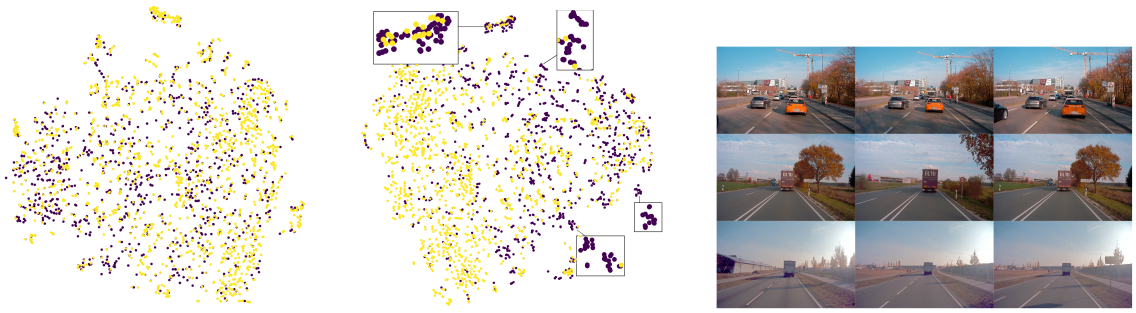
A	AL Method Metric →	SSL	Cityscapes: 2-2 mIoU@6	 AUC@10	PASCAL: 2-2 mIoU@6	 AUC@10
S	Random	✗	46.05	3.65	66.41	5.22
S	Entropy	✗	**51.24**	**4.00**	66.33	5.11
B	Coreset	✗	47.26	3.74	66.24	5.19
S	Random-SSL	✓	47.46	3.72	**68.60**	**5.37**
S	Entropy-SSL	✓	49.99	3.93	67.26	5.31
B	Coreset-SSL	✓	48.51	3.82	68.03	5.35
-	100%	✗	68.42	5.47	77.00	6.16

Table 3.8 Active Learning results on the PASCAL-VOC and Cityscapes dataset in low-budget 2-2 setting. AUC@10 and mIoU@6 metric are reported. A denotes Acquisition method type. S and B denotes the single-sample and batch-based acquisition, respectively.

A	AL Method Metric →	SSL	A2D2 Pool-0f 2-2 mIoU@6	 AUC@10	A2D2 Pool-11f 2-2 mIoU@6	 AUC@10
S	Random	✗	36.82	2.92	37.74	2.93
S	Entropy	✗	**41.40**	3.18	36.37	2.92
S	EqualAL	✗	41.13	**3.22**	37.28	2.97
B	Coreset	✗	40.18	3.12	**39.63**	**3.10**
S	Random-SSL	✓	37.80	2.99	36.46	2.90
S	Entropy-SSL	✓	38.32	3.03	36.70	2.93
S	EqualAL-SSL	✓	39.43	3.07	36.31	3.06
B	Coreset-SSL	✓	39.28	3.08	39.20	3.06
-	100%	✗	56.87	4.55	48.85	3.91

Table 3.9 Active Learning results on A2D2 Pool-0f in 2-2 setting. AUC@10 and mIoU@6 metrics are reported. A denotes Acquisition method type. S and B denotes the single-sample and batch-based acquisition, respectively.

of the new task is to select 3K images (similar size to Cityscapes) from the original A2D2 dataset ($\sim$40K images) to achieve the best performance. We select 3K images using active learning in 3 cycles with 1K images each. We compare 5 acquisition functions, including Random, Entropy, and Coreset, along with SSL integration. Such video datasets are often manually subsampled based on some prior information like time or location and then used for active learning. Therefore, we also include two such baselines - (a) where 3K samples are uniformly selected based on time information, denoted as Uniform, and (b) where every fifth sample is first selected uniformly to select $\sim 8K$ samples and then applied with Entropy acquisition function, denoted as Uniform(@5)+Entropy. The second approach is closer to previously used active learning benchmarks in the driving context. Results are shown in



(a) Coreset (Batch-based AL) (b) Entropy (Single-sample AL) (c) Mode collapse (Single-sample AL)

Fig. 3.13 TSNE plots of (a) Coreset and (b) Entropy functions for A2D2 Pool-21f. The yellow points are feature representation from the unlabeled set, the violet point are the acquired points. The batch-based approach has good selection coverage, whereas the single-sample acquisition approach selects similar samples from clusters. Figure (c) shows acquired redundant samples from the violet clusters in (b).

A	AL Method	SSL	mIoU	AUC
B	Uniform	✗	57.75	—
S	Random	✗	56.14	5.35
S	Entropy	✗	60.16	5.53
B	Coreset	✗	60.30	5.55
S	Uniform (@5) + Entropy	✗	60.40	5.66
B	Uniform-SSL	✓	58.93	—
S	Random-SSL	✓	57.57	5.53
S	Entropy-SSL	✓	59.91	5.61
B	Coreset-SSL	✓	**61.13**	**5.72**
S	Uniform (@5) + Ent-SSL	✓	59.63	5.59
-	100%	✗	66.65	6.64

Table 3.10 AL results on the proposed A2D2-3k task. mIoU@7.5 and AUC@7.5 are reported. S and B denote the single-sample and batch-based acquisition, respectively. Uniform refers to the temporal subsampling selection process and (@5) means every 5^{th} frame.

Table 3.10. We find that the batch-based Coreset-SSL method performs the best, discussed in Section 3.3.3, while the subsampling-based approaches are sub-optimal. This makes an excellent case for active learning in datasets with high redundancy, as active learning filters the data better than time-based subsampling methods.

3.3.5 A Polygon-based Annotation System

So far, the active learning methods for semantic segmentation were evaluated with the image-based annotation setting, where we assumed that all images have a equal annotation

cost and complete images were queried for annotation. In this section, we challenge the image-based annotation system with a polygon-based annotation system, where the image parts, in the form of polygons, can be queried and annotated.

Annotation model. A conventional active learning setup includes a human in the loop who annotates the samples picked by the query function. Since training with an actual human annotator is prohibitively expensive, we simulated its actions during training. We used the number of clicks required to annotate the entire image as a proxy for the human annotation cost. Assuming humans annotate by drawing polygons around each connected component in the image, we approximate each connected component in the ground truth image with a polygon using the Ramer-Douglas-Peucker algorithm [33]. The approximation quality is controlled by a pre-defined pixel-level tolerance parameter. The total number of clicks per image is then calculated by adding up the number of vertices for all polygons in this image. We perform a grid search over different tolerance values ranging from 5 to 40 pixels to find a suitable value. Figure 3.14 shows the trade-off between the average click cost per image and the polygon approximation quality of annotations for different tolerance values. The trade-off between different tolerance values and labeling quality is shown in Figure 3.15. Finally, we select the pixel-level labeling tolerance of 10 pixels. The approximated labels retain 95.06% mIoU as compared to the original ground-truth labels. According to this approximation, an average image costs around 33 clicks to label.

Experiment design. We show the performance of the AL methods for semantic segmentation on PASCAL-VOC 2012 [34]. The dataset consists of 20 foreground classes and one background class. We use the augmented annotated dataset, which contains 10582 training images and 1449 validation images.

In AL experiments for segmentation, we define the labeling cost in clicks. We use the initial labeling budget $\mathcal{B}_i$ and subsequent sampling budget $\mathcal{B}_s$ of 5000 clicks, which is approximately 1.5% of the total labeling cost of the dataset. In the first cycle, randomly sampled images are completely labeled until $\mathcal{B}_i$ is exhausted. In the subsequent cycles, an AL query method selects images based on a certain criterion and labels the picked image until $\mathcal{B}_s$ is exhausted. We test all the segmentation AL methods for 5 sampling cycles. All the results are shown on the validation set.

We evaluate AL methods for semantic segmentation in two different settings: (1) using standard augmentations and (2) using semi-supervised learning for training the target model. Our baseline methods include - Random selection, Entropy-based selection, Ensemble entropy-based selection, Learning loss-based selection, and SSL D-score selection. Random

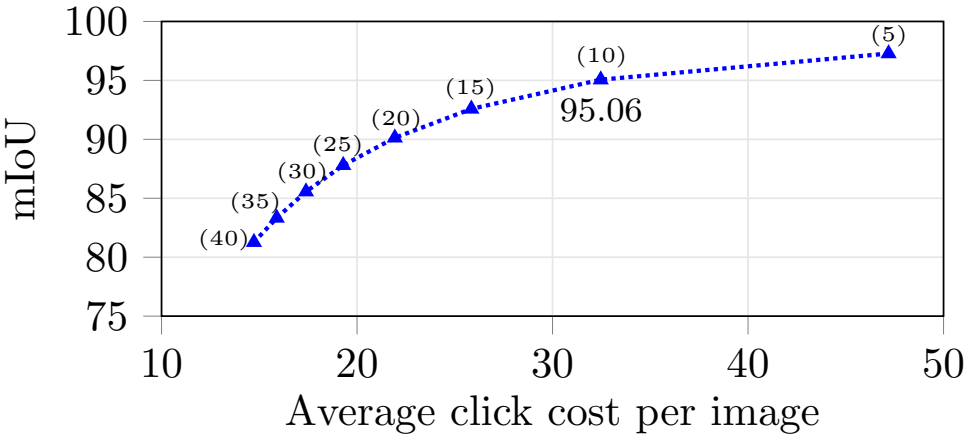


Fig. 3.14 Trade-off between the polygon approximation quality and the annotation cost in clicks. Tolerance values used for each measurement are mentioned above the ▲ markers.

and Entropy-based methods are already described in section 3.3.2. We describe the remaining methods below:

- *Ensemble with Average Entropy (ENS-ent):* This second uncertainty-based method ENS-ent is based on the average entropy over the predictions from all members of the model ensemble. We used the same information accumulation heuristic as used for the Entropy method.

- *Learning Loss (LL):* We adopted the LL [136] method from image classification to semantic segmentation. Since the original module is proposed for a Resnet architecture and the segmentation network used in this work is also based on a Resnet architecture, the exact method is directly adapted by reusing the original loss prediction module.

- *SSL-D-score:* Inspired by Ravanbakhsh *et al.* [100], we propose to use the discriminator of the s4GAN network as a query function for sampling. The output of the discriminator varies between 0 and 1, where a higher score is assigned to a higher quality of segmentation prediction. In other words, the discriminator of the s4GAN network acts as a critic, which provides a higher rating for better segmentation quality. This heuristic selects the samples which are not well represented by the current learned model, which is indicated with a lower rating. We refer to this semi-supervised approach for active learning as the SSL-D-score method.

The mean performance is reported over 3 trials for all single model-based methods and over 2 trials for ensemble-based methods due to higher computation costs.

Results: In the results, the uncertainty method based on entropy performs best and shows an improvement of around 1.1% mIoU over the random sampling baseline after 5

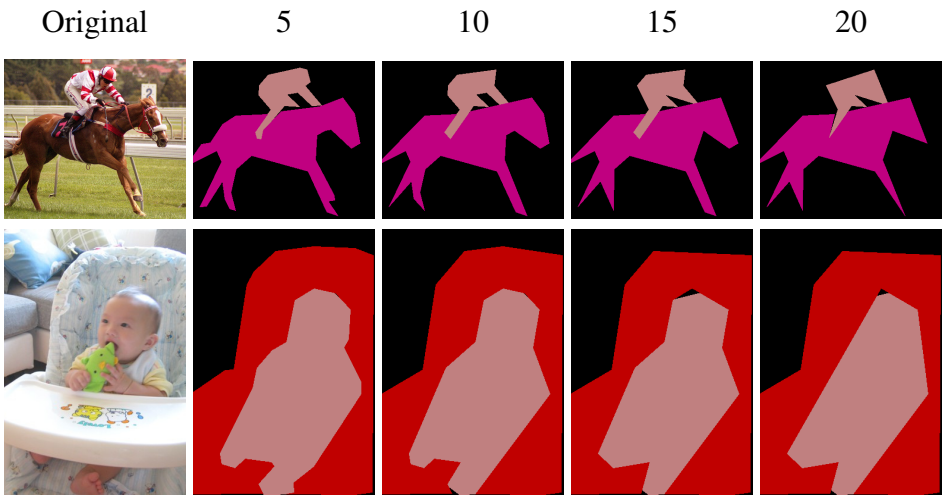


Fig. 3.15 Labeling quality when using polygon approximation with different tolerance values (in pixels). We picked a tolerance value of 10 pixels for our experiments.

AL sampling cycles. The LL method fails to outperform the random baseline approach. Corresponding performance curves are shown with solid lines in Figure 3.16.

The performance of the random baseline (Random), when combined with s4GAN, increases by the largest margin of 4.1% mIoU and reaches the overall best value. In addition, the SSL-D-score heuristic also shows comparable performance to the random baseline after 5 sampling cycles but does not bring any improvement over the SSL-Random baseline. The performance curves for all integrated methods are shown with dashed lines in Figure 3.16. Figure 3.18 shows the qualitative results at each sampling cycle, comparing the Entropy-Image method and the SSL-Random-Image baseline.

Polygon-based selection. In the above setting, labeling cost was defined per polygon, but the annotation was conducted image-wise. Here, we explore whether labeling only a part of an image is more effective than labeling the complete image. We evaluate active learning methods for semantic segmentation, where only a region of an image is selected by the query function. This region is approximately labeled using a polygon by the annotation simulator. We evaluate methods where an image is selected randomly, but the polygon in the image is selected based on the active learning heuristic. We compare entropy-based and random polygon selection methods in both raw and SSL-integrated active learning settings.

Experiment design: The entropy of a polygon is measured in a similar way as in the image-level labeling regime. We create a binary mask for the pixel-wise entropy based on a threshold and use the area of the high-entropy pixels as our selection heuristic. Only in the first cycle are images completely labeled until the $\mathcal{B}_i$ is covered. In the subsequent cycles,

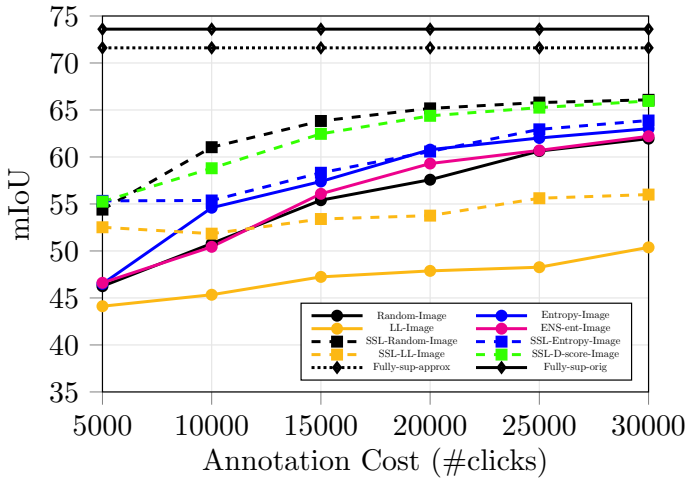


Fig. 3.16 Integrated SSL-AL methods for semantic segmentation mostly perform better than their raw counterparts on PASCAL-VOC with $\mathcal{B}_i = \mathcal{B}_s = 5000$ clicks ($\approx 1.5\%$ of the dataset). None of the methods outperforms SSL with random sampling.

images are labeled polygon-wise until the sampling budget $\mathcal{B}_s$ is exhausted. Figure 3.18 shows two examples of how the polygon-level labeling regime works based on the entropy heuristic. The budget settings and the hyperparameters exactly match those from the image-level labeling regime.

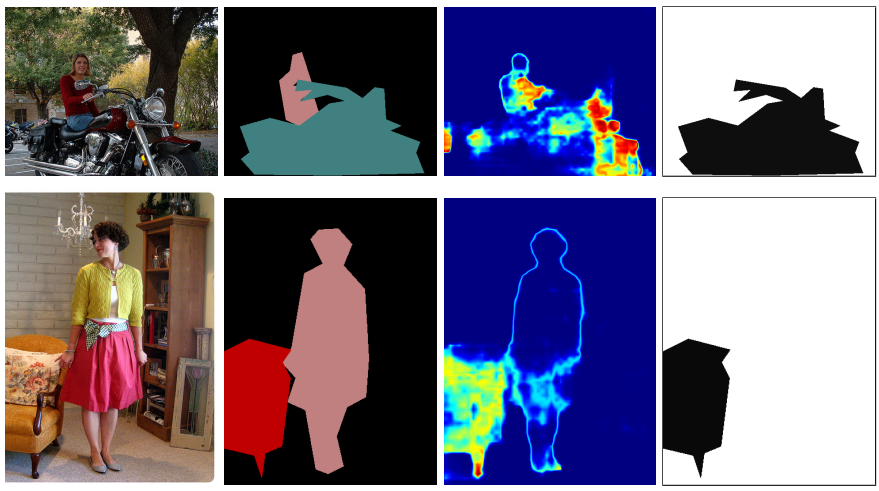


Fig. 3.18 Image labeling in a polygon-level labeling regime. From left to right: Original image, approximated ground-truth, pixel-wise entropy and selected polygon for labeling based on the entropy heuristic.

Results: Entropy-based polygon selection approach is more effective than random polygon selection for the raw active learning (without SSL) setting. However, when combined with semi-supervised learning, both entropy (Random-Image-Entropy-Polygon) and random

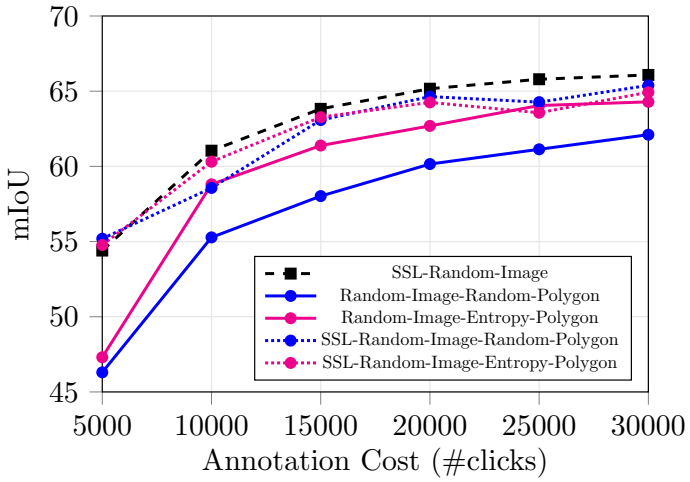


Fig. 3.17 Active learning for semantic segmentation: comparison between SSL integrated with active learning (SSL-X) against the standard setting. Results are shown on the PASCAL-VOC dataset with $\mathcal{B}_i = \mathcal{B}_s = 5000$ clicks. The suffixes 'Image' and 'Polygon' refer to the image-level and polygon-level labeling regimes respectively.

(Random-Image-Random-Polygon) polygon selection strategies perform very similarly. Results are shown in Figure 3.17. Moreover, when all polygon-level labeling approaches are compared with the image-level labeling approaches, we find SSL-Random-Image baseline even outperforms all the polygon-level active learning methods. In this experiment, we also observed that the SSL-Random-Image baseline outperformed the SSL-Random-Image-Random-Polygon baseline, showing that **image-level labeling is a more effective way of labeling an image**.

3.3.6 Conclusion

This work shows that active learning is indeed a useful tool for semantic segmentation. However, it is vital to understand the behavior of different active learning methods in various application scenarios. Table 3.11 provides an overview of the best-performing methods for each scenario for the semantic segmentation task. Our findings indicate that single-sample-based uncertainty is a suitable measure for sample selection in diverse datasets. In contrast, batch-based diversity-driven measures are better suited for datasets with high levels of redundancy. SSL is successfully integrated with batch-based diversity-driven methods. However, it can have a detrimental impact when combined with single-sample-based uncertainty acquisition functions. Active learning with a high annotation budget always performs better than random sampling and is further improved with the integration of

Dataset ↓	Annotation Budget			
	Low		High	
Sup. →	AL	SSL-AL	AL	SSL-AL
Diverse	Random	Random-SSL	Single	Single-SSL
Redundant	Batch	Batch	Batch	Batch-SSL

Table 3.11 Overview showing the best performing AL method for each scenario. Single and Batch refer to single-sample and batch-based method, and Random refers to random selection. Suffix -SSL refers to the usage of semi-supervised learning.

semi-supervised learning. As depicted in Figure 3.19, the batch-based methods are successful when there is a certain presence of redundancy in the dataset. Active learning with low annotation budgets is highly sensitive to the level of redundancy in the dataset. The optimal active learning policy changes from random selection to single-sample selection and then to batch-based selection based on the level of redundancy in the dataset. In this budget setting, SSL integration is only successful for highly diverse datasets. Our study, comparing polygon-based acquisition and image-based acquisition, shows that polygon-based selection does not offer any additional advantages over image-based acquisition. These findings have been missing in method development, which is usually optimized only for a few scenarios. The results of this study facilitate a broader view of the task with presumably positive effects in many applications.

3.4 Discussion

We observed that active learning acquisition methods for image classification are not very successful in outperforming naive random sampling when combined with some of the latest advances in the field. This makes us question the underlying issues present in the existing active learning literature for image classification. In order to stratify the issues found in this work, we provide a protocol to make the evaluation scheme more robust.

It would be interesting to know why AL often performs worse than random sampling for image classification and consistently does so in the low-budget regime. For now, we can only speculate. We believe that AL sampling introduces a bias into the distribution of annotated samples, i.e., the sampled distribution does not sufficiently match the true distribution anymore. The damage caused by this bias is more significant than the positive effect of learning from “more interesting” samples. If this hypothesis is true, research in active learning should focus on ways that find a way to control this harmful bias through the selection strategy.

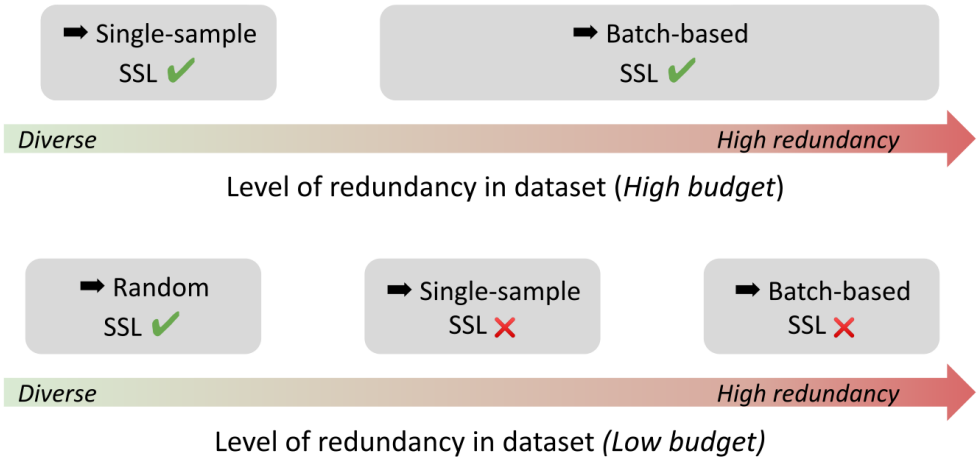


Fig. 3.19 The figure summarizes the results from our experiments. It shows how the ideal active learning policy changes w.r.t. the level of redundancy in the dataset given a high or low annotation budget. The first line in each block indicates the best acquisition objective, and the second line indicates whether SSL integration helps boost the performance of the best acquisition objective or not.

Results on semantic segmentation follow the trends observed in image classification, *i.e.* it is hard to pick one best method which performs well across all datasets and settings. However, we found it is possible to select the category of active learning acquisition method - single-sample based or batch-based, that can be effective if we can determine the nature of the dataset. Designing a metric that could identify the nature of the dataset as redundant or diverse is a non-trivial task and perhaps a direction to explore in future works.

In Chapters 2 and 3, we studied two approaches of learning from limited supervision, namely, semi-supervised learning and active learning. We analyzed existing active learning methods under realistic settings and also proposed new improved methods for semi-supervised learning. The overall goal of the approaches studied in these chapters was to improve label efficiency. In the next chapter, we will learn how to improve training efficiency in a continual learning setup, particularly in a class-incremental setup. In the next chapter, we identify novel underlying challenges in the continual learning solutions and propose simple solutions to rectify these issues which improve model performance.

Chapter 4

Class-incremental Continual Learning

The content of this chapter was adapted from the following paper.

> Sudhanshu Mittal, Silvio Galesso, and Thomas Brox. Essentials for class incremental learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pages 3513–3522, June 2021.

Silvio Galesso contributed by conducting the analysis of the effect of different regularizers on the feature representations in the above-mentioned paper. He co-designed and implemented evaluation metrics for measuring secondary class information. He also analyzed the expected calibration error for different regularizers. All co-authors contributed to the project discussions as well as the final paper text editing. All the other contributions described in this chapter are made by myself.

* * *

In the previous chapters, we learned about how to improve the label efficiency of deep learning models. In this chapter, we will focus on improving the training efficiency of deep learning models from a continual learning perspective. The objective of this chapter is to find and rectify the essential components that are important for deep learning models to continually learn from changing data distribution and tasks.

Motivation The ability to learn from continuously evolving data is important for many real-world applications. Traditional machine learning methods typically assume a fixed set of classes, which can be a limiting factor for a real-world application, where new classes emerge over time. For e.g., in a facial recognition system, the ability to adapt to new faces is important, or in autonomous driving systems, an agent might need to learn to recognize

and respond to new objects and situations on the road. Latest machine learning models, especially artificial neural networks, have shown great ability to learn the task at hand, but when confronted with a new task, they tend to override the previous concepts. This phenomenon of overriding and forgetting previous concepts is referred to as *catastrophic forgetting* in the literature. Deep neural networks suffer heavily from this catastrophic forgetting [84] when trained with a sequence of tasks, impeding continual or lifelong learning. Without a proper continual learning method, the system would need to be retrained on all previously seen data from all the tasks, which can be expensive and time-consuming. Several reasons have been attributed to this catastrophic forgetting issue, including - interference between the old and new information, overfitting to the new task, and limited capacity of the neural network. To mitigate catastrophic forgetting, several techniques have been proposed, including regularization, replay-based methods, and memory augmentation.

Problem setup. The problem of continual learning in the literature has been studied in numerous settings. They are defined depending on whether, at test time, the task identity is provided or not and whether the task identity needs to be inferred if it is not provided. This difference in experimental protocol influences the level of difficulty of the task. In this work, we study a continual learning setting, where models must be able to solve each task seen so far without any extra information about the test task identity. The problem is referred to as class-incremental learning (class-IL) [118]. It is one of the most difficult continual learning settings.

In class-incremental learning (class-IL) [101], the objective is to learn a unified classifier over incrementally occurring sets of classes. Since all the incremental data cannot be retained for unified training, the major challenge is to avoid forgetting previous classes while learning new ones. The other two scenarios in [118] include - (1) task incremental learning, where the output space of the tasks is different, and task-ID is provided during testing, and (2) domain-incremental learning, where the task remains the same but input distribution changes.

The three crucial components of a successful class-IL algorithm include a memory buffer to store a few exemplars from old classes, a forgetting constraint to keep previous knowledge while learning new tasks, and a learning system that balances old and new classes. Although several methods have been proposed to address each of these components, there is not yet a common understanding of best practices. In this work, we utilize previously proposed buffer memory and forgetting constraints but propose a novel, simple balanced learning system to reduce interference between old and new classes.

Problem definition. The objective of class-incremental learning (class-IL) is to learn a unified classifier from a sequence of data from different classes. Data arrives incrementally as a batch of per-class sets $\mathcal{X} = \{X^1, X^2, ..., X^t\}$, where X^y contains all images from class y. Learning from a batch of classes can be considered as a task $\mathcal{T}$. At each incremental step, the data for the new task $\mathcal{T}_i$ arrives, which exclusively contains samples of the new set of classes only. At each incremental step, data is only available for new classes $\mathcal{X}_{new} = \{X^{s+1}, ..., X^t\}$. Only a small amount of exemplar data $\mathcal{P}_{old} = \{P^1, ..., P^s\}$ from previous classes $\mathcal{X}_{old} = \{X^1, ..., X^s\}$ is retained in a memory buffer of limited size. Certain exemplar data is retained as a replay buffer to avoid complete forgetting. The model is expected to classify all the classes seen so far.

The problem definition with strictly separated batches may appear a bit specific. In many practical applications, the data will arrive in a more mixed-up fashion. However, this strict protocol allows the comparison of techniques, and it covers the key issues with class-incremental learning. Improvements on this protocol also serve less strict applied settings.

Scope of this chapter. In this chapter, we propose a compositional class-IL (CCIL) model that isolates the underlying reasons for catastrophic forgetting in class-IL and combines the most simple and effective components to build a robust base model. It employs plain knowledge distillation [50] as a forgetting constraint and selects exemplar samples simply randomly. For the loss evaluation, we propose important changes in output normalization. The goal of this part (Section 4.3) is to show that a balanced usage of simple components is sufficient to produce a strong model with state-of-the-art performance. In addition, we study the influence of the learned representation's properties on forgetting and show that the degree of feature specialization (overfitting) correlates with the degree of forgetting. We study some common regularization techniques and show that only those that keep, or even improve, the so-called secondary class information – also referred to as dark knowledge by [50] – have a positive influence on class-incremental learning, whereas others make things much worse. With these lessons learned, class-incremental learning results on CIFAR-100 and ImageNet improve over the state-of-the-art by a large margin while keeping the approach simple.

4.1 Related Work

iCaRL was the first approach that formally introduced the class-IL problem [101]. iCaRL is a decoupled approach for feature representation learning and classifier learning. It alleviates catastrophic forgetting via knowledge distillation and a replay-based approach. Later Castro

et al. [15] extended it to an end-to-end learning model based on a combination of distillation and cross-entropy loss to show improved results over iCaRL. Successive works usually dedicate their contribution to one of the three components in class-IL.

Exemplar selection: Replay-based approaches have been shown to be quite effective in mitigating catastrophic forgetting. Typically, a memory buffer is allocated to store exemplar samples of old classes, which are replayed while learning a new task to mitigate forgetting. Many works [15, 52, 101, 127] use herding heuristics [126] for exemplar selection. Herding selects and retains samples closest to the mean sample for each class. Liu *et al.* [79] parameterized the exemplars to optimize them jointly with the model. Iscen *et al.* [58] introduced a memory-efficient approach to store feature descriptors instead of images. In our work, we simply sample from each class randomly to compile the exemplar set.

Forgetting-constraint: Knowledge distillation (KD) was first introduced by Li *et al.* [74] for multi-task incremental learning. Thereafter, various works [15, 101, 127] have adopted it in class-IL to restore previous knowledge. Lately, several works have proposed new forgetting constraints with the objective of preserving the structure of old-class embeddings. Hou *et al.* [52] proposed the usage of feature-level distillation by penalizing the change in the feature representation from the old model. Yu *et al.* [138] utilized an embedding network to rectify the semantic drift, Tao *et al.* [115] proposed a Hebbian graph-based approach to retain the topology of the feature space. In this work, we utilize plain knowledge distillation, which is based on logits to avoid forgetting.

Bias removal methods: Various works [52, 127, 144] have pointed out that class-imbalance between old and new classes creates a bias in the class weight vectors in the last linear layer, due to which the network predictions are biased towards new classes. To rectify this bias, Wu *et al.* [127] trained an extra bias-correction layer using the validation set, Belouadah *et al.* [10] proposed to rectify the final activations using the statistics of the old task predictions, Zhao *et al.* [144] adjusted the norm of new class-weight vectors to those of the old class-weight vectors, and Hou *et al.* [52] applied cosine normalization in the last layer. The focus of these works is limited to the bias in the last layer, but ultimately catastrophic forgetting is an issue that affects the entire network: class imbalance causes the model to overfit to the new task, deteriorating the performance of the old ones. Some work [15, 73] also fine-tune the model to avoid overfitting to the current task. We propose a learning system that resolves this bias without the need for any post-processing by fixing the underlying issues; see Section 4.3.

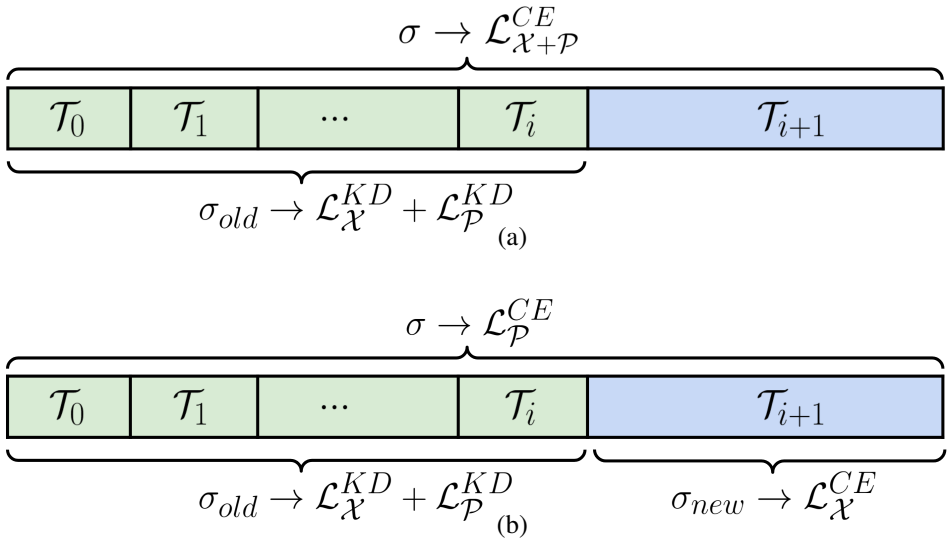


Fig. 4.1 The comparison between a (a) standard loss system and our proposed (b) compositional loss system (right). σ shows the softmax function span over all the network output logits. σ_{old} and σ_{new} show softmax span over the set of old and new class logits, respectively.

4.2 Basic Class-Incremental Learning Framework

The network model Θ consists of a feature extractor $\phi(\cdot)$ and a fully-connected layer $fc(\cdot)$ for classification. Similar to a standard multi-class classifier, the output logits o are processed through a softmax activation function $\sigma(\cdot)$ before cross-entropy loss $\mathcal{L}^{CE}$ is evaluated corresponding to the correct class. For the initial base task $\mathcal{T}_0$, the model Θ^s learns a standard classifier for the first ($y \in y[1:s]$) classes. In the incremental step, the fc layer is adapted to learn new classes ($y \in y[s+1:t]$) by adding new output nodes, whereas the other part of the network remains unchanged, resulting into a new model Θ^t. The three main elements of class-IL are set up as follows.

Exemplar selection. We compile the exemplar set by randomly selecting an equal number of samples (m) for each class. The samples are sorted in ascending order according to the distance from the mean of the feature vectors μ_i for each class separately. Since the size of the limited memory is fixed (K), some samples of old classes are removed to accommodate exemplars from new classes. Samples with larger distances to the mean vector are removed first. Detailed steps are shown in Algorithm 2.

Forgetting constraint. Our model uses knowledge distillation as the constraint against forgetting. Knowledge distillation penalizes the change with respect to the output of the old model (Θ^s) using KL-divergence, thus preserving the network's knowledge about the

old classes. The distillation loss ($\mathcal{L}^{KD}$) is computed for the exemplar sets ($\mathcal{P}$) as well as for samples from the new classes ($\mathcal{X}$). The final loss for our CCIL model is a combination of cross-entropy loss $\mathcal{L}^{CE}$ for classification and distillation loss $\mathcal{L}^{KD}$ for mitigating catastrophic forgetting as shown in Algorithm 1-Line 15.

Learning system. We propose a new compositional learning system that addresses the weight-bias issue in class-IL. The proposed loss isolates inter-task and intra-task learning for balanced processing of data by appropriately normalizing the output logits. The task-agnostic parts are shared to yield improved efficiency. The details are presented in Section 4.3.

4.2.1 Evaluation Metrics for Class-IL

Class-IL models are evaluated using three metrics: average incremental accuracy, forgetting rate, and feature retention. After each incremental step, all classes seen so far are evaluated using the latest model. After N incremental tasks, the accuracy $\mathcal{A}_n$ overall $(N+1)$ steps are averaged and reported. It is termed as *average incremental accuracy* (Avg Acc), introduced by Rebuffi *et al.* [101]. We also evaluate the *Forgetting Rate* $\mathcal{F}$ proposed by Liu *et al.* [79].

Algorithm 1: CCIL: IncrementalStep

Input: $\mathcal{X}_{new} = (X^{s+1}, ..., X^t), \mathcal{P}^s = (P_1, ..., P_s)$ // new classes data, old exemplar sets
Input: $K, \Theta^s, \hat{\Theta}^s$ // memory size, current model, frozen current model
Output: Θ^t // model trained on t classes

1 $m \leftarrow K/t$ // number of exemplars per class
2 $\Theta^t \leftarrow \Theta^s$ // add output nodes for new classes
3 $\mathcal{P} \leftarrow$ UpdateExemplarSets$(\mathcal{X}; \mathcal{P}^s, m, \Theta^s)$
4 **for** $(x, y) \in \mathcal{X}$ **do** // **update for mini-batch data in** $\mathcal{X}$
5 $o = \Theta^t(x)$ // $o = \{o_{old}, o_{new}\}$
6 softmax over new class logits $\sigma_{new}(o_{new})$
7 compute classification loss $\mathcal{L}^{CE}_{\mathcal{X}}$ (Eq. 4.3)
8 softmax over old class logits $\sigma_{old}(o_{old})$
9 compute distillation loss $\mathcal{L}^{KD}_{\mathcal{X}}$ (Eq. 4.4)
10 **load a mini-batch from exemplars set** $(x', y') \sim \mathcal{P}$
11 $o' = \Theta^t(x')$
12 softmax over all logits $\sigma(o')$
13 compute classification loss $\mathcal{L}^{CE}_{\mathcal{P}}$ (Eq. 4.5)
14 compute distillation loss $\mathcal{L}^{KD}_{\mathcal{P}}$ (Eq. 4.6)
15 $\mathcal{L} = (\mathcal{L}^{CE}_{\mathcal{X}} + \mathcal{L}^{CE}_{\mathcal{P}}) + \lambda * (\mathcal{L}^{KD}_{\mathcal{X}} + \mathcal{L}^{KD}_{\mathcal{P}})$
16 **end**

The forgetting rate measures the performance drop on the first task. It is the accuracy difference on the classes of the first task $X_{test}^{1:s}$, using Θ_0 and Θ_N. Θ_i denotes a model after i incremental steps. Therefore, it is independent of the absolute performance on the initial task $\mathcal{T}_0$. We introduce another metric, referred to as *Feature Retention* $\mathcal{R}_\phi$, to measure retention in the feature extractor $\phi(\cdot)$. It measures how much information is retained in the feature extractor while learning the tasks incrementally as compared to a jointly trained model. To measure $\mathcal{R}_\phi$: after the final incremental step, the parameters of the feature extractor are frozen, and the last linear layer is learned using all the data from all the classes. $\mathcal{R}_\phi$ is the accuracy difference between this model and a model where the whole network is trained on all the classes with complete data access. This metric measures the difference between the best possible features using joint training and features learned after incremental learning.

4.3 Compositional Learning System

For each gradient update, the CCIL model receives data in separate batches from the set of new classes $\mathcal{X}$ and the set of exemplars $\mathcal{P}$. $\mathcal{P}$ is the updated exemplar set which also includes the equal size of exemplars from the current new classes. (see Algorithm 1-Line 3) Instead of merging the batches, we propose to compute two separate losses for $\mathcal{X}$ and $\mathcal{P}$ mini-batches:

$$\mathcal{L}_\mathcal{X} = \mathcal{L}_\mathcal{X}^{CE} + \lambda * \mathcal{L}_\mathcal{X}^{KD} \tag{4.1}$$

$$\mathcal{L}_\mathcal{P} = \mathcal{L}_\mathcal{P}^{CE} + \lambda * \mathcal{L}_\mathcal{P}^{KD} \tag{4.2}$$

Intra-task learning. The classification loss for the new classes ($\mathcal{L}_\mathcal{X}^{CE}$) is computed using a dedicated softmax function σ_{new} comprising logits of new classes only (Figure 4.1b) computed as:

$$\mathcal{L}_\mathcal{X}^{CE} = -\sum_{i=s+1}^{t} y[i] \cdot \log(p_{new}[i]) \tag{4.3}$$

for $(x,y) \in \mathcal{X}$, where $p_{new} = \sigma_{new}(o_{new})$, $o = \Theta^t(x)$ and output logits comprise $o = \{o_{old}, o_{new}\}$. This allows the classifier weights for the new classes to be learned independently of the previous classes - while sharing the feature extractor, thus effectively eliminating the weight bias. Distillation loss ($\mathcal{L}_\mathcal{X}^{KD}$) is always computed using σ_{old} (see Figure 4.1b), since output of new network $p_{old} = \sigma_{old}(o_{old})$ are compared against the output of previous model

$\hat{p} = \sigma_{old}(\hat{\Theta}^s(x))$ as:

$$\mathcal{L}_{\mathcal{X}}^{KD} = D_{KL}(\hat{p}||p_{old}) \tag{4.4}$$

In the case of a unified softmax, the weights of the old classes are suppressed by the larger amount of new class samples during training. A similar analysis has been shown by [7] for a fine-tuning setup.

Inter-task learning. The separate softmax helps intra-task learning for the new classes, but this does not yet discriminate the new from the old classes. For inter-task learning, we plan a balanced interaction between the samples of old and new classes. We compile an exemplar set $\mathcal{P}$, which contains equal numbers of samples from all classes, including old and new classes. However small, such an exemplar set enables the model to capture the inter-task relationship through the loss $\mathcal{L}_{\mathcal{P}}^{CE}$, which uses a combined softmax function σ evaluated on all classes (see Figure 4.1b).

$$\mathcal{L}_{\mathcal{P}}^{CE} = -\sum_{i=1}^{t} y'[i] \cdot \log(q[i]) \tag{4.5}$$

for $(x', y') \in \mathcal{P}$, where $q = \sigma(o')$ and $o' = \Theta^t(x')$. The distillation loss is computed similarly to Eq. 4.4,

$$\mathcal{L}_{\mathcal{P}}^{KD} = D_{KL}(\hat{q}||q_{old}) \tag{4.6}$$

where $\hat{q} = \sigma_{old}(\hat{\Theta}^s(x'))$ and $q_{old} = \sigma_{old}(o'_{old})$. This exemplar set is compiled before learning the incremental task, contrary to previous works, where it is always compiled after the incremental step. Figure 4.1 shows how the loss terms are calculated using a separate softmax function (Figure 4.1b) and also compares it to the unified softmax (Figure 4.1a) used in previous works.

Transfer learning. We observed that a separate softmax does not remove the bias completely. Another cause for unbalanced class-weight vectors, and catastrophic forgetting in general, is the change in the data distribution between different tasks. We hypothesize that the effect of this distribution shift in the training data is more harmful to the previous knowledge when the transfer learning from old to new classes is poor, resulting in a strong alteration of the parameters of the network. We propose to reduce the learning rate for the incremental steps as a simple way to improve transfer learning and mitigate the adverse effect of the distribution shift. This further helps reduce the weight bias. The reduced learning rate on

Algorithm 2: UpdateExemplarSets

Input: $\mathcal{X}, \mathcal{P}_{old}$ // new class data, old exemplar set
Input: Θ^s, m // old model, new exemplar size per class
Output: $\mathcal{P}_{new}$ // new Exemplar sets

```
for i = 1,...,s do
    P_i ← (p_1,...,p_m) // keep first m samples
end
/* add new class exemplars                                   */
for i = s+1,...,t do
    P_i ← (p_1,...,p_m) ⊂ X^i) // randomly pick m samples
    μ_i ← (1/m) Σ_{j=1}^{m} φ(p_j) // mean feature
    /* sort exemplars based on distance from μ_i             */
    for k = 1,...,m do
        p_k ← arg min ||μ_i − φ(p_k)||
    end
end
```

incremental steps depends on the scale and relevance of features learned in the base task; therefore, it is determined experimentally. Although lowering the learning rate is a standard technique when fine-tuning a network on a new dataset, its importance is underestimated and often missing in incremental learning works. We experimentally show its importance in ablation studies (Section 4.5.2).

4.4 Improving Feature Representations for Incremental Learning

Intuitively, poorly transferable embeddings will force the model to alter its parameters significantly in order to learn new concepts. This destroys the knowledge accumulated for the previous tasks. In this section, we explore this novel direction- aiming to learn robust representations that are transferable to a new task and effectively retain previous knowledge in class-IL. In particular, we study the detrimental effects of overfitting and loss of secondary class information. We find that: 1) both phenomena strongly correlate with catastrophic forgetting; 2) regularization methods can significantly improve robustness against forgetting, but only as long as they enhance the secondary class information of the learned model.

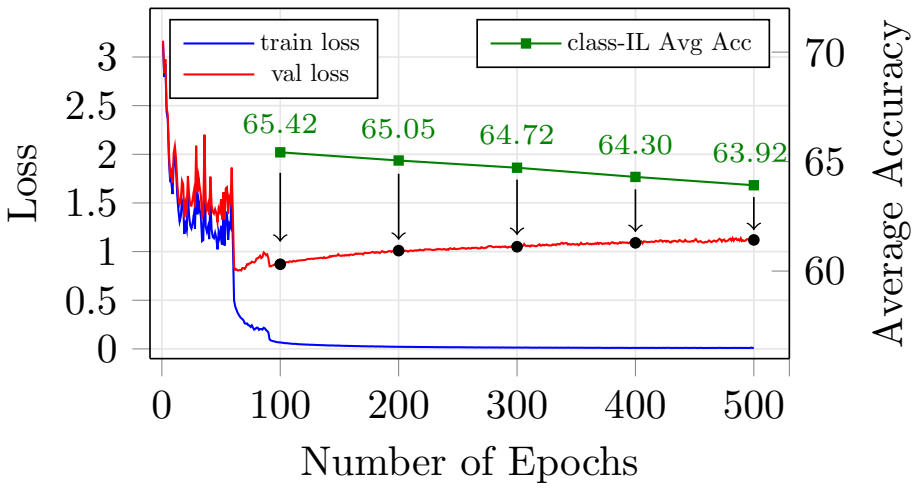


Fig. 4.2 The effect of overfitting on class-IL performance on the CIFAR-100 dataset. The figure shows the overfitting behavior on the initial base task. The validation loss (red curve) starts increasing monotonically after the 100^{th} epoch. The green curve shows the average incremental accuracy (right y-axis) for class-IL experiments performed over different snapshots at every 100^{th} epoch.

4.4.1 Measuring the Quality of Secondary Logits

Secondary information captures the semantic relationship between the target and non-target classes. In literature, the term *secondary information* is interchangeably used to denote the non-target and non-maximum scores of a classifier [134]. Here, for evaluation purposes, the term denotes the non-maximum scores produced by the networks. When applying the maximum operation to the scores predicted by a classifier, part of the information produced by the model is discarded. For each individual sample, this information represents the model's belief about the semantic nature of the image in relation to the other classes. It is important to learn this secondary information such that the model can re-use it to learn new classes with the least modification to previous concepts. We argue that semantically similar classes should lie closer in the representation space as compared to the dissimilar classes since they share more features, and higher secondary information is an indicator of such an efficient non-redundant feature space. We have included an analysis on feature representations to support this argument in Section 4.5.4.

No proper annotations exist for secondary information; therefore, we define a proxy evaluation objective, exploiting the *coarse*-labeling of the CIFAR-100 dataset, which partitions the 100 *fine*-classes into 20 superclasses. The 5 classes belonging to each superclass are mostly semantically related and have been previously used for evaluating secondary information [134]. As a proxy evaluation measure for secondary class information, we propose to use the classification performance on the superclasses, restricting the network

Epoch	**SS-NLL** $\downarrow$	**SS-Acc** $\uparrow$	**Avg Acc** $\uparrow$	$\mathcal{F}\downarrow$	$\mathcal{R}_\phi\downarrow$	ECE
100	2.54 ± 0.04	38.68 ± 0.89	65.42 ± 0.06	16.03 ± 0.36	9.04 ± 0.24	0.093 ± 0.003
200	2.89 ± 0.06	32.88 ± 0.59	65.05 ± 0.08	16.04 ± 0.26	9.27 ± 0.42	0.118 ± 0.003
300	3.03 ± 0.06	30.09 ± 0.53	64.72 ± 0.07	16.94 ± 0.61	9.51 ± 0.23	0.126 ± 0.004
400	3.09 ± 0.07	29.04 ± 0.68	64.3 ± 0.12	18.38 ± 0.19	9.68 ± 0.17	0.131 ± 0.005
500	3.11 ± 0.03	27.97 ± 0.54	62.92 ± 0.11	18.57 ± 0.39	10.00 ± 0.20	0.137 ± 0.002

Table 4.1 The effect of overfitting on class-IL performance and its correlation with secondary information. Table shows the performance of the network snapshots taken at every 100^{th} epoch. Accuracy decreases and SS-NLL increases, both monotonically, as more severely overfitted models are evaluated. Forgetting rate $\mathcal{F}$ also correlates with overfitting. Results are computed over 5 runs.

output to the non-maximum logits. We define two new metrics for this purpose: Secondary Superclass NLL and Secondary Superclass Accuracy.

Secondary Superclass-NLL (SS-NLL). Negative Log Likelihood is a commonly used cost function for classification, also known as *Cross-Entropy Loss*. Here we compute the NLL induced by the secondary (non-maximum) logits on the superclass classification problem. Given a set of superclasses $\mathcal{S}$, we can group the fine-grained classes into subsets $\mathcal{C}$ according to their coarse-label, and compute:

$$\textit{SS-NLL}(x,y) = -\sum_{j\in\mathcal{S}}\left[\mathbb{1}_{\mathcal{C}_j}(y)\log\sum_{k\in\mathcal{C}_j}\hat{\sigma}_k\big(f(x)\big)\right], \tag{4.7}$$

where $\mathbb{1}_{\mathcal{C}_j}(y)$ is an indicator function which evaluates to 1 if the true class y belongs to superclass j, $\hat{\sigma}$ is a softmax function over the secondary fine-logits (i.e. it suppresses the maximum logit). The network prediction (logits) is denoted as $f(x)$. A lower SS-NLL indicates better superclass classification and, thus, higher secondary information quality.

Secondary Superclass-Accuracy (SS-Acc): Secondary superclass accuracy computes the percentage of correct superclass predictions. As for SS-NLL, the largest logit score is excluded from the prediction to focus the measure on the quality of secondary information. Higher SS-Acc values indicate higher quality of the secondary information.

4.4.2 Forgetting starts before the incremental step

In this section, we study how the quality of the representations learned during the initial base task correlates with incremental learning performance. We experimentally show how a decline in the quality of the learned features - measured as overfitting and loss of secondary

information - leads to higher catastrophic forgetting, motivating our following search for a suitable regularizer.

Experiment details: We set up a standard class-IL experiment (with 5 incremental tasks) on CIFAR-100, using a ResNet-32 model. The initial base network is trained for up to 500 epochs. We employ an SGD optimizer with a base learning of 1e-1, weight decay of 5e-4, and momentum 0.9. We use a step learning rate schedule, where the learning rate is divided by 10 at 60^{th} and 90^{th} epochs.

Analysis: Figure 4.2 shows that the validation loss (red curve) starts increasing after about 100 epochs, showing an overfitting effect. Thereafter, we perform five different class-IL experiments, each based on a different snapshot of the base network (every 100^{th} epoch). As the validation loss of the snapshot increases, incremental learning performance of the corresponding class-IL model drops (green curve), and both forgetting rate ($\mathcal{F}$) and feature retention metric ($\mathcal{R}_\phi$) worsen (Table 4.1). The worsening $\mathcal{R}_\phi$ metric indicates that the issue is rooted in the feature representations, and cannot be mitigated by acting on the last layer bias. Along with these metrics, we observe that overfitting causes the quality of secondary information to deteriorate (SS-Acc decreases and the SS-NLL increases, Table 4.1). This loss of secondary information could also be linked to increasing overconfidence of the network, measured as Expected Calibration Error (ECE) [47]. Table 4.1 also shows the expected calibration error (ECE) with standard deviation for different snapshots of the overfitted model. It shows that ECE monotonically increases with the number of training epochs.

These results indicate that: (1) the quality of the features learned during the first base task influences the performance of the class-IL model, and as such, it should be expressly addressed. (2) secondary information can be considered as an indicator of the features' quality and their fitness for incremental learning. In the next section, we will show experimental evidence in support of these hypotheses.

4.4.3 Analyzing Catastrophic Forgetting with Regularization

Having established a link between early feature quality and catastrophic forgetting, we hypothesize that the application of adequate regularization techniques can improve model performance on the task at hand. We apply four common regularization techniques to our CCIL model: self-distillation [37], data-augmentation (including cropping, cutout [32], and an extended set of AutoAugment [26] policies), label smoothing [113], and mixup [142]. All these regularizers have been shown to improve generalization on the held-out validation data.

Model	**Avg. Acc.↑**		**SS Metrics**		$\mathcal{F}\downarrow$	$\mathcal{R}_\phi\downarrow$	**ECE↓**
	5 tasks	10 tasks	SS-NLL ↓	SS-Acc. ↑			
CCIL	66.44 ± 0.31	64.86 ± 0.40	2.784	34.83	17.13	9.70	0.100
CCIL + SD	67.17 ± 0.14	65.86 ± 0.29	2.675	37.26	16.81	8.88	0.094
CCIL + H-Aug	71.66 ± 0.23	69.88 ± 0.36	2.051	47.69	13.37	6.73	0.018
CCIL + LS	63.08 ± 0.21	61.99 ± 0.30	3.103	24.25	18.79	12.83	0.049
CCIL + Mixup	62.31 ± 0.46	57.75 ± 1.64	2.791	31.57	24.56	16.01	0.024

Table 4.2 Effect of regularization class-IL average accuracy, secondary information (on the first-task model), forgetting rate and feature retention (5 tasks), on CIFAR-100. All the values are averaged over 3 runs. ↓ and ↑ in the column headings indicate that lower and higher values are better respectively. Values that are better than the CCIL baseline are marked in green whereas the worse ones are marked in red. SD:self-distillation, LS:label-smoothing, H-Aug:heavy data augmentation.

Self-distillation. Self-distillation [37, 88] is a form of knowledge distillation in which the teacher and student networks have the same architecture. It can be applied iteratively, in generations: at each generation, a copy of the current student becomes the new teacher, with proven positive effects on generalization.

Data augmentation. Augmentation is one of the most widespread regularization techniques for neural networks, especially in computer vision. A well-designed data augmentation routine is key to obtaining good results on the held-out dataset. We sample randomly from a pool of augmentation policies that contain pairs of different geometric and color transformations, similarly to [26].

Label smoothing. Label smoothing [113] acts on the cross-entropy loss for classification by interpolating the one-hot labels with a uniform distribution over the possible classes. This technique has been shown to improve generalization and reduce overconfidence of classification models [113].

Mixup. Mixup [142] is an operation that generates training samples for classification by linearly combining pairs of existing samples - image and label. Mixup has successfully been used as a form of data augmentation in image classification, improving generalization and calibration [142, 117].

Analysis. We analyze above discussed metrics for each of these regularization techniques. Table 4.2 shows the Average Accuracy after finishing the last incremental step, secondary information quality of the first task model, forgetting rate, feature retention (Section 4.2.1)

and expected calibration error [47]. We can divide the regularization methods into two groups: the ones which improve class-IL performance (self-distillation, augmentation) and the ones which harm it (label smoothing, mixup). The first group also shows consistent improvements in secondary information and reduction in forgetting, with augmentation performing the best across all metrics - by a significant margin. In the second group, label smoothing harms secondary information the most. It has been observed that label smoothing encourages representations to be closer to their respective class centroid and equidistant to the other class centroids [90], and this comes at the expense of inter-class sample relationships, i.e., secondary information. Mixup also harms the quality of secondary information: we believe this is because it artificially forces arbitrary distances between classes, which modifies the natural output distribution - similarly to label smoothing. Interestingly, all regularizers improve network calibration, but ECE is not a good indicator of class-IL performance, unlike secondary information, shown in Table 4.2.

In summary, label smoothing and mixup - despite their proven regularization effects - harm secondary class information and have clear negative consequences for class-incremental learning. On the other hand, regularization methods that enhance secondary class information (self-distillation and data augmentation) boost the average incremental accuracy. Analogously to the analysis of Section 4.4.2, we show that the quality of secondary information negatively correlates to the forgetting rate (Table 4.2), further indicating the importance of secondary class information.

4.5 Experiments and Results

4.5.1 Training Details

Datasets. We conduct experiments on CIFAR100 [69], ImageNet-100 Subset [28] and full ImageNet datasets. CIFAR-100 contains 60K images from 100 classes of size 32×32, with 50K images for training and 10K for evaluation. The ImageNet-100 dataset has 100 randomly sampled classes (using Numpy seed:1993) from ImageNet. The base CCIL model uses default data augmentation, including random cropping and horizontal flipping for CIFAR-100 and resized-random cropping and horizontal flipping for ImageNet datasets. All the randomization seeds are selected following the experiments in previous works [52, 79].

CIFAR-100 classes are shuffled using a fixed seed (Numpy [119] seed:1993) across all methods for a fair comparison. The ImageNet-100 dataset has 100 randomly sampled classes (using Numpy seed:1993) from ImageNet and further shuffled (using Numpy seed:1993). It

contains around 128K images of size 224×224 for training and 5K images for evaluation. ImageNet-1k classes are also shuffled using a Numpy seed:1993.

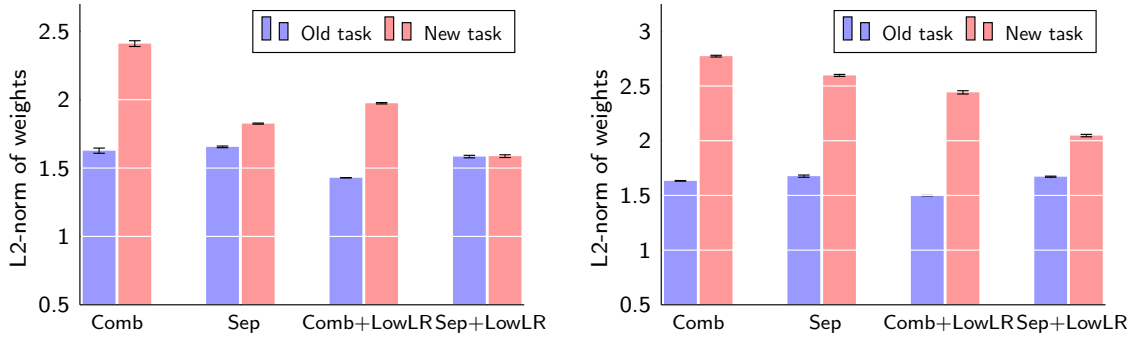
Benchmark protocol. We follow the protocol used in previous works [52, 79]. The protocol involves learning 1 initial base task followed by N incremental tasks. We evaluate with two incremental settings: where the model learns $N = 5$ and $N = 10$ incremental tasks. For CIFAR-100 and ImageNet-100, 50 classes are selected as the base classes for the initial task, and the remaining classes are equally divided over the incremental steps. A similar format is followed for ImageNet with 500 base classes. Exemplar memory size is set to $K = 2k$ for 100 class datasets and $K = 20k$ for the full ImageNet dataset.

Implementation details. We use a 32-layer ResNet [49] for CIFAR-100 dataset, and a 18-layer ResNet for ImageNet-100 and ImageNet datasets. The last layer is cosine normalized following the recommendations of [52]. On CIFAR-100, the base network is trained for 120 epochs using a cosine learning rate schedule, where the base learning rate is 1e-1. Subsequent N tasks are trained for 240 epochs with a base learning rate of 1e-2. The learning rate is decayed until 1e-4. We use a batch size of 100 for CIFAR-100 experiments. Networks for the CIFAR-100 dataset are optimized using the SGD optimizer with a momentum of 0.9 and weight decay of 5e-4. For ImageNet-100, the network is trained for 70 epochs using a step learning rate schedule, where the base learning rate is 1e-1 for the base task and 1e-2 for the subsequent N tasks. The base learning rate is divided by 10 at $\{30, 60\}$ epochs. For ImageNet, the base task is trained for 70 epochs following a step learning rate, where the base learning is 1e-1. The base learning rate is divided by 10 at $\{30, 60\}$ epochs. The incremental task is trained for 40 epochs following a step learning rate, where the base learning rate starts from 1e-2. The base learning rate is divided by 10 at $\{25, 35\}$ epochs. Networks for ImageNet datasets are optimized using the SGD optimizer with a momentum of 0.9 and weight decay of 1e-4. We use a batch size of 128 for both ImageNet datasets.

For *self-distillation* experiments, self-distillation is conducted over 4 generations (optimized using validation performance) for CIFAR-100 and ImageNet-100 datasets and over 2 generations for the ImageNet dataset. At the beginning of each self-distillation generation, the network snapshot (student) becomes the teacher network, and the student continues to train (fine-tuned) with a combination of classification and distillation loss. For CIFAR-100, the self-distillation model is trained for 70 epochs with a decaying (cosine) learning rate from 1e-1 to 1e-3. All other optimizer settings are the same as the baseline model. For ImageNet-100, the self-distillation model is trained for 30 epochs each, where the base learning rate is 1e-2, and it is divided by 10 at 10, 20 epochs. For ImageNet, the self-distillation model is

Operations	Avg Acc ↑ w/o KD	Avg Acc ↑ w/ KD
Comb	47.97	52.71
Sep	52.86	60.85
Comb+LowLR	52.79	54.54
Sep+LowLR	**58.60**	**64.79**

Table 4.3 Table contains the corresponding class-IL results without distillation (w/o KD) and with distillation (w/ KD) in terms of average incremental accuracy. All experiments use the linear classification layer. Results shown on CIFAR-100 for 5-task experiments.



(a) Weight norm w/o KD

(b) Weight norm w/ KD

Fig. 4.3 (a) & (b) compares the average L_2 norm of the classification weight vectors for old and new classes for class-IL experiments without (w/o) and with (w/) KD respectively. We evaluate standard combined softmax (Comb) against proposed separate softmax (Sep) and we assess the effect of reduced learning rate (LowLR).

trained for 15 epochs each, where the base learning rate is 1e-2, and it is divided by 10 at 8, 12 epochs.

4.5.2 Ablation Studies

Elements of the compositional learning system. We evaluate the contributions of each element in the proposed learning system by training multiple class-IL models featuring them. The incremental learning in these experiments is conducted in two settings - in a simple fine-tuning setup (without distillation) in order to single out the effects of the proposed changes and with knowledge distillation loss included. In Figure 4.3a & 4.3b, we compare the average L2 norm of the class weight vectors for old and new classes after 5 incremental training steps, while in Table 4.3 we provide the average accuracies of the respective models. We notice a major difference in the weight norms of old and new classes

Method	**Layer**		**Softmax**		**Low LR**	**AW**	**Classifier**		**KD**	**Avg Acc**
	Cos	Dot	Sep	Comb			NME	CNN		
Comb		✓		✓				✓		47.97
iCaRL		✓		✓			✓		✓	56.50
iCaRL++	✓			✓		✓		✓	✓	59.78
CCIL	✓		✓		✓	✓		✓	✓	66.44

Table 4.4 Drawing parallels between iCaRL and our proposed model. Average accuracy is reported for 5-task class-IL experiments on the CIFAR-100 dataset. The last row highlights our proposed changes. All methods use random exemplar selection as used in this work, Dot: linear layer, KD: knowledge distillation, NME: nearest-mean-of-exemplars (used in [101])

for the default combined softmax (Comb) setting (Figure 4.1a). Using separate-softmax (Sep) substantially reduces this difference and improves class-IL performance but does not resolve the problem completely. A lower learning rate (Comb+LowLR) also removes the bias and improves the performance, although to a lesser extent. When both approaches are combined (Sep+Low-LR), this bias is largely resolved, and the best class-IL results are produced.

Drawing parallels with iCaRL. We compare different components of our CCIL model with the first baseline approach (iCaRL) proposed by [101]. Table 4.4 summarizes these changes. We first isolate the contributions of some follow-up methods by creating another baseline as iCaRL++. It consists of a (1) cosine-normalized layer (cos) [44, 81, 52], where the features and class-weight vectors in the final layer are normalized to lie in a high-dimensional sphere. It helps in removing the remaining weight bias during inference, and (2) adaptive weighting (AW), where the weight of the distillation loss increases with incremental steps. AW was previously introduced in [52], which helps in the adaptive balancing of classification and distillation loss. The adaptive weighting function λ (similar to [52]) between two losses is defined as:

$$\lambda = \lambda_{base} \left(\frac{C_n + C_o}{C_n} \right)^{2/3} \tag{4.8}$$

,where C_n denotes number of new classes, C_o denotes number of old classes, λ_{base} is fixed constant for each method. It dynamically increases weightage on preserving old knowledge as incremental training continues. It improves the baseline model by 0.45% for 5 task experiments on CIFAR-100. $\lambda_{base} = 5$ is set for CIFAR-100, $\lambda_{base} = 20$ for ImageNet-100 and $\lambda_{base} = 600$ for ImageNet. The last row in Table 4.4 shows that replacing the combined-

Method	CIFAR-100		ImageNet-100		ImageNet	
No. of i-tasks →	5	10	5	10	5	10
iCaRL* [101]	57.17	52.57	65.04	59.53	51.50	46.89
BIC [127]	59.36	54.20	70.07	64.96	62.65	58.72
WA [144]	63.25	58.57	—	—	—	—
LUCIR [52]	63.12	60.14	70.47	68.09	64.34	61.28
Mnemonics [79]	63.34	62.28	72.58	71.37	64.54	63.01
TPCIL [115]	65.34	63.58	76.27	74.81	64.89	62.88
CCIL (ours)	66.44	64.86	77.99	75.99	67.53	65.61
CCIL-SD (ours)	**67.17**	**65.86**	**79.44**	**76.77**	**68.04**	**66.25**
Joint-training	74.12	73.80	84.72	84.67	69.72	69.75

Table 4.5 Comparing average incremental accuracy computed using different methods on CIFAR-100, ImageNet-100 and ImageNet dataset. *as reported in [52]

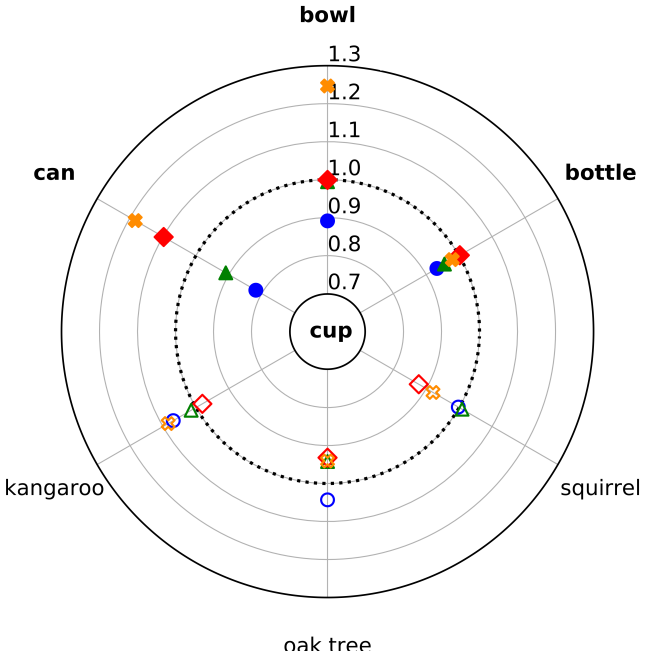
softmax (comb) with the proposed separate-softmax (sep) and reducing the learning rate (LowLR) yields a major improvement.

4.5.3 Comparison to SOTA

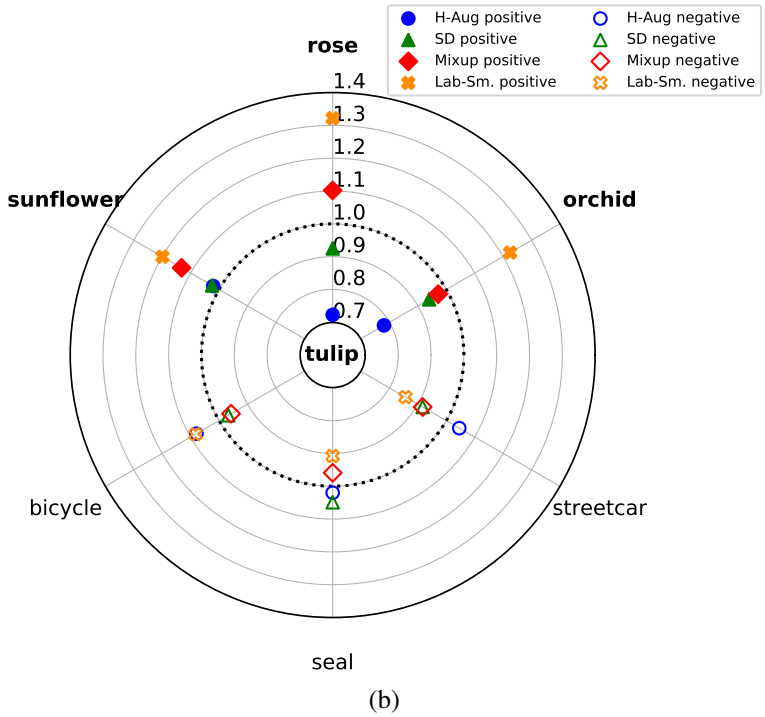
Results for CIFAR-100, ImageNet-100, and ImageNet datasets are shown in Table 4.5. We report the upper bound 'Joint-training', where at every incremental step, all the data for the classes seen until then is accessible. The simple CCIL model compares favorably to previous results on all datasets, especially on larger datasets like ImageNet-1k. The regularized CCIL-SD closes the gap to joint training further and achieves state-of-the-art performance across all datasets. Since the CCIL model is based only on simple components, we believe that the application of advanced methods for mitigating forgetting [52, 115] and more informative exemplar selection [79] can further improve the performance.

4.5.4 Representations: Qualitative Analysis

This section provides a qualitative analysis on the effect of different regularizers on the feature representations (penultimate-layer activations). We analyze the representations of the network trained on 50 classes (first task) of the CIFAR-100 dataset using the ResNet-32 network.



(a)



(b)

Fig. 4.4 Effect of regularizers on the distance between mean class representations. The numbers shown in the plot are the ratios between the class means distances of each method and of the default CCIL model. Similar classes are marked in **bold**. *Dotted* circle at 1.0 depicts distances between classes in the baseline CCIL model and other distances are depicted relative to the baseline model. *Positive* and *negative* cases indicate similar and dissimilar classes respectively.

Class-mean Representations: We argue that the classes which are semantically similar must be closer in the representation space as compared to the dissimilar classes since they share more features. Based on this argument, we analyze the effect of different regularization methods on the relative distances between class-mean representations. We utilize the fine- and coarse-label structure of the CIFAR-100 dataset to compare the effect on the distance between semantically similar and dissimilar classes relative to the default baseline model. Classes associated with the same coarse label or superclass are considered as similar classes, whereas dissimilar classes are picked from different superclasses. L2 distance is used as the distance metric.

Figure 4.4 show this qualitative analysis for two classes: *cup* and *tulip*. For example, cup and can are semantically similar classes. When self-distillation and augmentation are used as regularizers, the relative distance reduces to 0.9 and 0.8, respectively, whereas when label-smoothing and mixup are applied, the relative distance increases to 1.2 and 1.1, respectively. Other similar classes follow a similar trend, whereas dissimilar pairs show opposite behavior. Overall we find that regularizers: self-distillation, and heavy data augmentation reduce the relative distance between similar classes (marked in bold) while not affecting or increasing the distance between dissimilar classes. Whereas mixup and label smoothing increase the relative distance between similar classes and reduce the relative distance between dissimilar classes. We notice that these observations agree with the findings on secondary class information presented in Section 4.4.3.

We also argued that label-smoothing and mixup regularization deteriorate secondary class information since they dismantle the natural output distribution. This qualitative analysis supports our argument showing how they conversely hamper the distances between similar and dissimilar classes.

4.6 Summary

In this chapter, we presented a straightforward class-incremental learning system that focuses on the essential components and already exceeds the state of the art without integrating sophisticated modules. The proposed compositional model is motivated by the findings on imbalanced class weight vectors. It isolates and recombines differently affected components in the network to build an improved model. Since final model is a simple and effective solution, this makes it a good base model for future research on advancing class-incremental learning.

Moreover, we showed that countering catastrophic forgetting during the incremental step is not enough: the quality of the feature representation prior to the incremental step

considerably determines the amount of forgetting. We empirically displayed that the degree of overfitting on the prior task correlates with the degree of forgetting in the incremental learning steps. This suggests that representation learning is a promising direction to maximize also incremental performance. In this regard, we showed that boosting secondary information is key to improving the transferability of features from old to new tasks without forgetting.

Chapter 5

Conclusion

In this thesis, we focused on two aspects of improving the efficiency of the deep learning models. In Chapter 2 and 3, we learned how to best utilize manual supervision while keeping the annotation cost as low as possible. In Chapter 4, we provided new insights and methodology toward making continual learning models successful while keeping the requirements of training resources low.

In Chapter 2 of the thesis, we introduced a novel semi-supervised learning approach for semantic segmentation that is based on a generative adversarial network. Our approach is designed to be end-to-end learnable and one-stage, which is stable across different settings and benchmarks. Our proposed model is a two-branch model which aims to rectify artifacts at different levels. It leverages an online self-training approach that helps stabilize the generator and discriminator of the GAN model and enhances the model's predictive performance. One major contribution is the use of the Feature Matching loss, which is also crucial for a stable adversarial training process. Another strength of our approach is its versatility, as it can utilize various forms of extra supervision, such as image-level labels and scribbles. The effectiveness of the method is shown qualitatively and quantitatively on three segmentation benchmarks. Our method also performs competitively well compared to the latest state-of-the-art SSL methods.

In Chapter 3 of the thesis, we question the existing evaluation practices used in deep active learning for image classification and semantic segmentation tasks and seek answers to the specific missing questions that previous works have failed to capture. First, we identify that the deep AL methods for image classification are often tested under incompatible conditions: different architectures, different augmentation strategies, budgets, etc., and often ignoring the latest parallel works in the field, like semi-supervised learning and strong data augmentation. In this work, we study active learning methods under the influence of data augmentation and semi-supervised learning across different annotation budgets. Our findings indicate that

these additional techniques help in improving the model performance in most cases but often fail to improve consistently over a simple random baseline. For the semantic segmentation task, we explored the influence on existing active learning methods across three dimensions - data distribution w.r.t. different redundancy levels, integration of semi-supervised learning, and different labeling budgets. We found that these three underlying factors are crucial for selecting the best active learning acquisition function. We observe that the ideal active learning policy changes from a single-sample selection objective to a batch-based selection objective as the level of redundancy in the dataset increases. Integration of semi-supervised learning is not always helpful, as demonstrated in the case of image classification, but rather depends on the data distribution and the objective of the active learning acquisition function. Integration of semi-supervised learning is consistently supported for batch-based acquisition methods and redundant datasets. Moreover, we demonstrate that the selection of the best AL policy requires precise knowledge about the underlying training conditions when the available annotation budget is quite low due to its highly volatile nature.

In the last chapter, we propose a simple yet effective solution for class-incremental learning systems that focuses on the essential components and already outperforms the state-of-the-art without integrating sophisticated modules. We analyze the cause of catastrophic forgetting in such continual learning setups and provide a straightforward compositional model that addresses the issue to a great extent. Our proposed model combines inter-task and intra-task learning components using the model architecture and objective function in an informed manner. We further shed light upon other factors that contribute to catastrophic forgetting, such as overfitting. We found that an overfitted model lacks the ability to learn new tasks without losing its previous knowledge. This suggests that a transferable representation is crucial for the success of a class-incremental learning system. We show that the secondary class information is a good proxy measure for the transferability of the representation. We believe that it is a promising future direction to seek regularization techniques that enhance secondary class information captured by the network.

Future Work

Despite the recent rapid progress in making deep learning more efficient, many challenges still remain open and highly relevant. Here, we provide a few most important directions for future research.

Open-vocabulary semantic segmentation. Due to the recent success of large vision-language models, the idea of learning with limited supervision has been transformed into open-vocabulary prediction, where any objects can be retrieved or classified just using a

text query. Such vision-language models are usually trained on millions of image-text pairs and are capable of solving open-world tasks like image classification and object detection. Such image-text pairs are much easier and cheaper to retrieve from the internet compared to manual annotations. Recently, open-vocabulary semantic segmentation has also gained interest, where the objective is to segment any class in the image based on a text query. Although it currently falls short of supervised baseline performance, we believe this is a promising future direction to create open-world models without much manual annotation expenses.

Meta-policy for active learning. In our work on deep active learning, we showed that different active learning policies are suited for different data distributions w.r.t. the level of redundancy in the dataset. When the given dataset has a high level of redundancy, a diversity-driven batch-based approach is more suitable, whereas when the dataset is very diverse, an uncertainty-driven per-sample-based approach is a better approach. Firstly, one must measure the redundancy levels in the dataset to select the ideal policy. Defining such a redundancy measure is a logical next step in this direction. Secondly, as the data annotation process via the active learning cycle goes on, the nature of the dataset changes, for *e. g.* , when collecting annotated data from a raw video for a specific application. Initially, the diversity-driven policy is ideal due to the high redundancy in the dataset. However, after collecting sufficient annotated data, finding special cases where the model is most uncertain is essential. Therefore, we need a meta-policy that can switch between different acquisition objectives depending on the model's current state.

References

References [1–6] are included in the list of my publications at the beginning of the thesis.

[7] Hongjoon Ahn, Jihwan Kwak, Subin Lim, Hyeonsu Bang, Hyojun Kim, and Taesup Moon. Ss-il: Separated softmax for incremental learning. *arXiv preprint arXiv:2003.13947*, 2020.

[8] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*, 2018.

[9] Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. There are many consistent explanations of unlabeled data: Why you should average. In *ICLR*, 2019.

[10] Eden Belouadah and Adrian Popescu. Il2m: Class incremental learning with dual memory. In *IEEE International Conference on Computer Vision*, 2019.

[11] William H. Beluch, Tim Genewein, Andreas Nürnberger, and Jan M. Köhler. The power of ensembles for active learning in image classification. In *CVPR*, 2018.

[12] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. In *ICML*, 2015.

[13] Gabriel J. Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 2008.

[14] Lile Cai, Xun Xu, Jun Hao Liew, and Chuan Sheng Foo. Revisiting superpixels for active learning in semantic segmentation with realistic annotation costs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[15] Francisco M. Castro, Manuel J. Marin-Jimenez, Nicolas Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[16] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien, editors. *Semi-Supervised Learning*. The MIT Press, 2006.

[17] Baixu Chen, Junguang Jiang, Ximei Wang, Pengfei Wan, Jianmin Wang, and Mingsheng Long. Debiased self-training for semi-supervised learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[18] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *CoRR*, abs/1412.7062, 2014.

[19] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2018.

[20] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018.

[21] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *CoRR*, abs/1802.02611, 2018.

[22] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2613–2622, June 2021.

[23] Noel C. F. Codella, David Gutman, M. Emre Celebi, Brian Helba, Michael A. Marchetti, Stephen W. Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, and Allan Halpern. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi). In *IEEE 15th International Symposium on Biomedical Imaging (ISBI)*, 2018.

[24] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[25] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[26] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, 2019.

[27] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, 2015.

[28] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[29] Jia Deng, Wei Dong, Richard Socher, Li jia Li, Kai Li, and Li Fei-fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[30] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[31] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 2012.

[32] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[33] David H. Douglas and Thomas K. Peucker. Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature. In *Classics in Cartography*. 2011.

[34] Mark Everingham, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010.

[35] Mark Everingham, Luc van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 2010.

[36] Geoff French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations, 2020.

[37] Tommaso Furlanello, Zachary Chase Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born-again neural networks. In *International Conference on Machine Learning*, 2018.

[38] Yarin Gal and Zoubin Ghahramani. Bayesian convolutional neural networks with Bernoulli approximate variational inference. In *ICLR-workshop track*, 2016.

[39] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *ICML*, 2016.

[40] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *ICML*, 2017.

[41] Mingfei Gao, Zizhao Zhang, Guo Yu, Sercan Ö. Arık, Larry S. Davis, and Tomas Pfister. Consistency-based semi-supervised active learning: Towards minimizing labeling cost. In *ECCV*, 2020.

[42] Xavier Gastaldi. Shake-shake regularization. *arXiv preprint arXiv:1705.07485*, 2017.

[43] Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S. Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, Tiffany Fernandez, Martin Jänicke, Sudesh Mirashi, Chiragkumar Savani, Martin Sturm, Oleksandr Vorobiov, Martin Oelker, Sebastian Garreis, and Peter Schuberth. A2D2: Audi Autonomous Driving Dataset. 2020.

[44] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[45] Kitani Kris Golestaneh, S. Alireza. Importance of self-consistency in active learning for semantic segmentation. *BMVC*, 2020.

[46] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*. 2014.

[47] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

[48] Bharath Hariharan, Pablo Arbelaez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, 2011.

[49] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[50] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[51] Seunghoon Hong, Hyeonwoo Noh, and Bohyung Han. Decoupled deep neural network for semi-supervised semantic segmentation. In *NIPS*. 2015.

[52] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[53] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *ArXiv*, abs/1112.5745, 2011.

[54] Hanzhe Hu, Fangyun Wei, Han Hu, Qiwei Ye, Jinshi Cui, and Liwei Wang. Semi-supervised semantic segmentation via adaptive equalization learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 22106–22118. Curran Associates, Inc., 2021.

[55] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018.

[56] Siyu Huang, Tianyang Wang, Haoyi Xiong, Jun Huan, and Dejing Dou. Semi-supervised active learning with temporal output discrepancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[57] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. In *BMVC*, 2018.

[58] Ahmet Iscen, Jeffrey Zhang, Svetlana Lazebnik, and Cordelia Schmid. Memory-efficient incremental learning through feature adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[59] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, 2018.

[60] S. D. Jain and K. Grauman. Active image segmentation propagation. In *CVPR*, 2016.

[61] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gürel, Bo Li, Ce Zhang, Dawn Song, and Costas J. Spanos. Towards efficient data valuation based on the shapley value. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, 2019.

[62] Elmer H. Johnson. Elementary applied statistics: For students in behavioral science. *Social Forces*, 1966.

[63] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *NeurIPS*, 2017.

[64] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *CVPR*, 2017.

[65] Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. Glister: Generalization based data subset selection for efficient and robust learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

[66] Kwanyoung Kim, Dongwon Park, Kwang In Kim, and Se Young Chun. Task-aware variational adversarial active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[67] Diederick P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[68] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, 2019.

[69] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[70] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *ICLR*, 2017.

[71] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *NeurIPS*. 2017.

[72] Yann Lecun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998.

[73] Kibok Lee, Kimin Lee, Jinwoo Shin, and Honglak Lee. Overcoming catastrophic forgetting with unlabeled data in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[74] Z. Li and D. Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[75] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, 2016.

[76] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, 2017.

[77] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014.

[78] B. Liu and V. Ferrari. Active learning for human pose estimation. In *ICCV*, 2017.

[79] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[80] Pauline Luc, Camille Couprie, Soumith Chintala, and Jakob Verbeek. Semantic segmentation using adversarial networks. In *NIPS Workshops*. 2016.

[81] Chunjie Luo, Jianfeng Zhan, Xiaohe Xue, Lei Wang, Rui Ren, and Qiang Yang. Cosine normalization: Using cosine similarity instead of dot product in neural networks. In *International Conference on Artificial Neural Networks*, 2018.

[82] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.

[83] Radek Mackowiak, Philip Lenz, Omair Ghori, Ferran Diego, Oliver Lange, and Carsten Rother. Cereals - cost-effective region-based active learning for semantic segmentation. In *BMVC*, 2018.

[84] Michael Mccloskey and Neil J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *The Psychology of Learning and Motivation*, 1989.

[85] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high-and low-level consistency. *IEEE transactions on pattern analysis and machine intelligence*, 43(4):1369–1379, 2019.

[86] Sudhanshu Mittal, Maxim Tatarchenko, Özgün Çiçek, and Thomas Brox. Parting with illusions about deep active learning. *CoRR*, abs/1912.05361, 2019.

[87] Takeru Miyato, Shin ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *TPAMI*, 2018.

[88] Hossein Mobahi, Mehrdad Farajtabar, and Peter L. Bartlett. Self-distillation amplifies regularization in hilbert space. *arXiv:12002.05715*, 2020.

[89] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014.

[90] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? In *Advances in Neural Information Processing Systems*, 2019.

[91] Prateek Munjal, Nasir Hayat, Munawar Hayat, Jamshid Sourati, and Shadab Khan. Towards robust and reproducible active learning using neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[92] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *NeurIPS*. 2018.

[93] Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. In *NeurIPS*. 2016.

[94] George Papandreou, Liang-Chieh Chen, Kevin P. Murphy, and Alan L. Yuille. Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *ICCV*, 2015.

[95] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[96] R. Paul, D. Feldman, D. Rus, and P. Newman. Visual precis generation using coresets. In *ICRA*, 2014.

[97] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel matters – improve semantic segmentation by global convolutional network. In *CVPR*, 2017.

[98] Pedro O. Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015.

[99] Aneesh Rangnekar, Christopher Kanan, and Matthew Hoffman. Semantic segmentation with active semi-supervised representation learning, 2022.

[100] Mahdyar Ravanbakhsh, Tassilo Klein, Kayhan Batmanghelich, and Moin Nabi. Uncertainty-driven semantic segmentation through human-machine collaborative learning. In *International Conference on Medical Imaging with Deep Learning – Extended Abstract Track*, 2019.

[101] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[102] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In *NIPS*. 2016.

[103] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

[104] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Annual Workshop on Computational Learning Theory*, 1992.

[105] Claude Elwood Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 1948.

[106] Gyungin Shin, Weidi Xie, and Samuel Albanie. All you need are a few pixels: Semantic segmentation with pixelpick. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, 2021.

[107] Changjian Shui, Fan Zhou, Christian Gagné, and Boyu Wang. Deep active learning: Unified and principled method for query and training. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, 2020.

[108] Amber L. Simpson, Michela Antonelli, Spyridon Bakas, Michel Bilello, Keyvan Farahani, Bram van Ginneken, Annette Kopp-Schneider, Bennett A. Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M. Summers, Patrick Bilic, Patrick F. Christ, Richard K. G. Do, Marc Gollub, Jennifer Golia-Pernicka, Stephan H. Heckers, William R. Jarnagin, Maureen K. McHugo, Sandy Napel, Eugene Vorontsov, Lena Maier-Hein, and M. Jorge Cardoso. A large annotated medical image dataset for the development and evaluation of segmentation algorithms. Workingpaper, February 2019.

[109] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[110] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 596–608. Curran Associates, Inc., 2020.

[111] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In *ICCV*, 2017.

[112] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958, 2014.

[113] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[114] Meng Tang, Federico Perazzi, Abdelaziz Djelouah, Ismail Ben Ayed, Christopher Schroers, and Yuri Boykov. On regularized losses for weakly-supervised cnn segmentation. In *ECCV*, 2018.

[115] Xiaoyu Tao, Xinyuan Chang, Xiaopeng Hong, Xing Wei, and Yihong Gong. Topology-preserving class-incremental learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[116] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*. 2017.

[117] Sunil Thulasidasan, Gopinath Chennupati, Jeff A. Bilmes, Tanmoy Bhattacharya, and Sarah Michalak. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. In *Advances in Neural Information Processing Systems*, 2019.

[118] Gido M. van de Ven and Andreas S. Tolias. Three scenarios for continual learning. *arXiv:1904.07734*, 2019.

[119] S. van der Walt, S. C. Colbert, and G. Varoquaux. The numpy array: A structure for efficient numerical computation. *Computing in Science Engineering*, 2011.

[120] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Trans. Cir. and Sys. for Video Technol.*, 2017.

[121] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Trans. Cir. and Sys. for Video Technol.*, 2017.

[122] Shuo Wang, Yuexiang Li, Kai Ma, Ruhui Ma, Haibing Guan, and Yefeng Zheng. Dual adversarial network for deep active learning. In *ECCV*, 2020.

[123] Yuchao Wang, Haochen Wang, Yujun Shen, Jingjing Fei, Wei Li, Guoqiang Jin, Liwei Wu, Rui Zhao, and Xinyi Le. Semi-supervised semantic segmentation using unreliable pseudo labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[124] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *CVPR*, 2017.

[125] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S. Huang. Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation. In *CVPR*, 2018.

[126] Max Welling. Herding dynamical weights to learn. In *International Conference on Machine Learning*, 2009.

[127] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[128] Zifeng Wu, Chunhua Shen, and Anton van den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 90:119–133, 2019.

[129] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 12077–12090. Curran Associates, Inc., 2021.

[130] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*, 2019.

[131] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves imagenet classification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[132] Haiming Xu, Lingqiao Liu, Qiuchen Bian, and Zhen Yang. Semi-supervised semantic segmentation with prototype-based consistency regularization. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[133] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. *arXiv preprint arXiv:2202.11094*, 2022.

[134] Chenglin Yang, Lingxi Xie, Siyuan Qiao, and Alan L. Yuille. Knowledge distillation in generations: More tolerant teachers educate better students. In *Association for the Advancement of Artificial Intelligence*, 2019.

[135] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z. Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *Medical Image Computing and Computer Assisted Intervention - MICCAI*, 2017.

[136] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *CVPR*, 2019.

[137] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. BDD100K: A diverse driving video database with scalable annotation tooling. *CoRR*, abs/1805.04687, 2018.

[138] Lu Yu, Bartlomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic drift compensation for class-incremental learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[139] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[140] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016.

[141] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. In *CVPR*, 2018.

[142] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[143] Yizhe Zhang, Michael T. C. Ying, Lin Yang, Anil T. Ahuja, and Danny Z. Chen. Coarse-to-fine stacked fully convolutional nets for lymph node segmentation in ultrasound images. In *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2016.

[144] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[145] Fedor Zhdanov. Diverse mini-batch active learning. *CoRR*, abs/1901.05954, 2019.

[146] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019.