DCU

Ollscoil Chathair
Bhaile Átha Cliath
Dublin City University

DUBLIN CITY UNIVERSITY
SCHOOL OF ELECTRONIC ENGINEERING

Deep learning for computer vision constrained by limited supervision

PAUL ALBERT M.E.

Dissertation submitted in fulfilment of the requirements for the award of Doctor of Philosophy (PhD)

SUPERVISED BY DR. KEVIN MCGUINNESS
AND PROF. NOEL E. O'CONNOR

February 2023

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work, and that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others and to the extent that such work has been cited and acknowledged within the text of my work.

Signed : Paul Albert ID number: 19212190

Date : 1^{st} of August 2023

Acknowledgments

I learned on this PhD journey that good research is never the result of one brilliant first author but a collective collaboration of ideas from the group.

I would start by issuing my special thanks to my closest co-authors Eric Arazo and Diego Ortego who have instilled the research spirit and ethics into my work. The seed you planted 4 years ago is still growing and will continue to for many years! You both are the researchers I looked up to at the beginning of this PhD and I hope that my contributions have lived up to the research quality you showed me how to produce.

To my first supervisor Kevin McGuinness, thank you for your reassurance and a firm shoulder when I needed to explore some experimental findings further in the unwavering Tuesday meetings. Conducting a PhD during the COVID pandemic was challenging at times but I am grateful we were able to catch up in person over our join Hawaii conference attendance. To Noel O'Connor, none of this would have been possible without you. You took a chance on me 5 years ago on a 5 months internship in the Insight lab. Thank you for the advice sessions we had and for making time in your busy schedule for a chat whenever I needed one. To both Kevin and Noel, thank you for making travel funds available to travel to conferences when my budget ran out. Attending in person, even when on the other side of the planet, was very valuable to discover new research areas and to build relations as a researcher. I am very grateful that you made this possible.

Some special thanks to my colleagues in the Vistamilk SFI centre: Mohamed Saadeldin I highly value the very practical advice you gave me in person and in our bi-weekly meetings. Deidre Hennessy, thank you for putting in the work to make our collaboration move forward, paring computer vision and grassland researchers was no easy task. I believe that the research showcased in this thesis will demonstrate we made it happen anyways. To Donagh Berry, I learned the important values of a research centre leader from you. Your proximity with the staff and desire to always improve collaborations in the centre are in great part responsible for the success of VistaMilk.

Thank you to Tarun Krishna for comments and regular suggestions on paper reads in the weekly meetings, to Luis Lebron Casas and Enric Moreu for the research related discussions or not we had over the last 4 years. I wish you all continued success for the remainder of your PhD and the future. You are all in good hands but you know that already!

To my parents Isabelle and Laurent, I feel very lucky and grateful to have had so many opportunities handed to me by default. I wonder could I have made it without them? Having special support from scientist parents is always rewarding when pursuing a scientific career. Special thoughts to my three brothers at home: Quentin, Benoit and Théo and to my grand-parents André, Marie-Thérèse, Jean-Claude and Claudie. Hopefully this thesis will explain some of the obscure work I have been up to in these last 4 years.

I will finish with my Petronela for your patience and support be it in Dublin or down south in Enniscorthy. Multă dragoste.

Publications

First author peer-reviewed papers:

- **Paul Albert**, Eric Arazo, Tarun Krishna, Noel O'Connor, Kevin McGuinness. "Is your noise correction noisy? PLS: Robustness to label noise with two stage detection." In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. January 2023.

- **Paul Albert**, Eric Arazo, Noel O'Connor, Kevin McGuinness. "Embedding contrastive unsupervised features to cluster in- and out-of-distribution noise in corrupted image datasets." In *IEEE/CVF European Conference on Computer Vision (ECCV)*. October 2022.

- **Paul Albert**, Mohamed Saadeldin, Badri Narayanan, Brian Mac Namee, Deidre Hennessy, Noel E. O'Connor, Kevin McGuinness. "Utilizing unsupervised learning to improve sward content prediction and herbage mass estimation." In *29th European Grassland Federation (EGF) General Meeting*. June 2022.

- **Paul Albert**, Mohamed Saadeldin, Badri Narayanan, Brian Mac Namee, Deidre Hennessy, Noel E. O'Connor, Kevin McGuinness. "Unsupervised domain adaptation and super resolution on drone images for autonomous dry herbage biomass estimation." In *IEEE/CVF Conference on Compuer Vision and Pattern Recognition Workshops (CVPRW)*. June 2022.

- **Paul Albert**, Diego Ortego, Eric Arazo, Noel O'Connor, Kevin McGuinness. "DSOS: Addressing out-of-distribution label noise in webly-labelled data." In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. January 2022.

- **Paul Albert**, Mohamed Saadeldin, Badri Narayanan, Brian Mac Namee, Deidre Hennessy, Aisling H. O'Connor, Noel E. O'Connor, Kevin McGuinness. "Semi-supervised dry herbage mass estimation using automatic data and synthetic images." In *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. October 2021.

- **Paul Albert**, Diego Ortego, Eric Arazo, Noel O'Connor, Kevin McGuinness. "ReLaB: Reliable Label Bootstrapping for Semi-Supervised Learning." In *International Joint Conference on Neural Networks (IJCNN)*. July 2021.

Secondary author peer-reviewed papers:

- Badri Narayanan, Mohamed Saadeldin, **Paul Albert**, Kevin McGuiness, Brian Mac Namee. "Adaptation of Compositional Data Analysis in Deep Learning to Predict Pasture Biomass Proportions." In *Irish Conference on Artificial Intelligence and Cognitive Science (AICS)*. December 2021.

- Eric Arazo, Diego Ortego, **Paul Albert**, Noel O'Connor, Kevin McGuinness. "How Important is Importance Sampling for Deep Budgeted Training?" In *International Joint Conference on Neural Networks (IJCNN)*. July 2021.

- Diego Ortego, Eric Arazo, **Paul Albert**, Noel O'Connor, Kevin McGuinness. "Multi-Objective Interpolation Training for Robustness to Label Noise." In *Computer Vision and Pattern Recognition (CVPR)*. June 2021.

- Deirdre Hennessy, Mohamed Saad, Brian Mac Namee, Noel O'Connor, Kevin McGuinness, **Paul Albert**, Badri Narayanan and Aisling O'Connor, "Using image analysis and machine learning to estimate sward clover content," In *European Grassland Federation (EGF)*. May 2021.

- Diego Ortego, Eric Arazo, **Paul Albert**, Noel O'Connor, Kevin McGuinness. "Towards Robust Learning with Different Label Noise Distributions." In *International Conference on Pattern Recognition (ICPR)*. January 2021.

- Eric Arazo, Diego Ortego, **Paul Albert**, Noel O'Connor, Kevin McGuiness. "Pseudo-labeling and confirmation bias in deep semi-supervised learning." In *International Joint Conference on Neural Networks (IJCNN)*. July 2020.

- Badri Narayanan, Mohamed Saadeldin, **Paul Albert**, Kevin McGuiness, Brian Mac Namee. "Extracting Pasture Phenotype and Biomass Percentages using Weakly Supervised Multi-target Deep Learning on a Small Dataset." In *Irish Machine Vision and Image Processing (IMVIP)*. July 2020.

- Eric Arazo*, Diego Ortego*, **Paul Albert**, Noel O'Connor, Kevin McGuinness. "Unsupervised label noise modeling and loss correction." In *International Conference on Machine Learning (ICML)*. June 2019.

(*) Equal contribution

The code for reproducing results in these papers and this thesis can be found at `github.com/PaulAlbert31?tab=repositories`.

Contents

List of Figures

List of Tables

Notations

EMA	Exponential Moving Average
MLP	Multi-Layer Perceptron
CNN	Convolutional Neural Network
AUC	Area Under the Curve
kNN	k-nearest neighbor classifier
GPU	Graphics Processing Unit
$\boldsymbol{\mathcal{D} = \{(x_i, y_i)\}^N}$	An image dataset of size N where each image x_i in the dataset is associated to a label y_i.
$\boldsymbol{ip}$	Inner product
$\boldsymbol{C}$	Number of classes in a classfication dataset.
$\boldsymbol{h}$	A classification neural network.
$\boldsymbol{\Phi}$	A regression neural network.
$\boldsymbol{\Psi}$	A semantic segmentation neural network.
$\boldsymbol{l_{ce}\left(x_i, y_i\right) = -\, y_i^T \log h(x_i)}$	The cross-entropy loss.
$\boldsymbol{\hat{y}_i}$	The estimated label for the image x_i.
OOD	Out-of-distribution
ID	In-distribution
DSOS	Dynamic Softening of Out-of-distribution Samples
SNCF	Spectral Noise clustering from Contrastive Features
ReLaB	Reliable Label Bootstrapping
DM	Dry herbage matter
TPR	True positive rate
FPR	False positive rate

Metrics

The metrics in this thesis are typically computed over N observations. In the follwing table y denotes the ground-truth classification label while $\hat{y}$ is the prediction made by the neural network.

Name	Acronym	Units	Formula
Classification accuracy	–	%	$\sum_1^N \frac{1}{N}\mathbb{1}_{(\hat{y}=y)}$
Classification error	–	%	$\sum_1^N \frac{1}{N}\mathbb{1}_{(\hat{y}\neq y)}$
Area Under the Curve	AUC	–	Area under the TPR vs FPR curve
Dry Herbage composition RMSE	RMSE	%	$\sqrt{\sum_1^N \frac{1}{N}(\hat{y}-y)^2}$
Dry Herbage mass RMSE	HRMSE	$kg.DM.ha^{-1}$	$\sqrt{\sum_1^N \frac{1}{N}(\hat{y}-y)^2}$
Grass Height RMSE Error	HE	cm	$\sqrt{\sum_1^N \frac{1}{N}(\hat{y}-y)^2}$
Herbage Relative Absolute Error	HRAE	%	$\sum_1^N \frac{1}{N}\frac{\lvert y-\hat{y}\rvert}{\hat{y}}$
Dry Herbage Relative Error	HRE	–	$\sum_1^N \frac{1}{N}\frac{\hat{y}}{y}$

Abstract

"Deep learning for computer vision constrained by limited supervision"
Paul Albert

This thesis presents the research work conducted on developing algorithms capable of training neural networks for image classification and regression in low supervision settings. The research was conducted on publicly available benchmark image datasets as well as real world data with applications to herbage quality estimation in an agri-tech scope at the VistaMilk SFI centre. Topics include label noise and web-crawled datasets where some images have an incorrect classification label, semi-supervised learning where only a small part of the available images have been annotated by humans and unsupervised learning where the images are not annotated. The principal contributions are summarized as follows. Label noise: a study highlighting the dual in- and out-of-distribution nature of web-noise; a noise detection metric than can independently retrieve each noise type; an observation of the linear separability of in- and out-of-distribution images in unsupervised contrastive feature spaces; two noise-robust algorithms DSOS and SNCF that iteratively improve the state-of-the-art accuracy on the mini-Webvision dataset. Semi-supervised learning: we use unsupervised features to propagate labels from a few labeled examples to the entire dataset; ReLaB an algorithm that allows to decrease the classification error up to 8% with one labeled representative image on CIFAR-10. Biomass composition estimation from images: two semi-supervised approaches that utilize unlabeled images either through an approximate annotator or by adapting semi-supervised algorithms from the image classification litterature. To scale the biomass to drone images, we use super-resolution paired with semi-supervised learning. Early results on grass biomass estimation show the feasibility of automating the process with accuracies on par or better than human experts. The conclusion of the thesis will summarize the research contributions and discuss thoughts on future research that I believe should be tackled in the field of low supervision computer vision.

Chapter 1

Introduction

Section 1.1 introduces the motivations behind the research carried out in this thesis, Section 1.2 proposes research hypothesis and research questions. Section 1.3 presents the structure of the report.

1.1 Motivations and Research Hypothesis

Deep learning is the state-of-the-art approach to solve computer vision tasks yet the high accuracy results reported in the literature are bound to the human supervision required to curate and annotate datasets. Reducing the amount of human supervision, particularly in terms of data annotation and curation needed to produce accurate deep learning models is necessary to enable the deployment of state-of-the-art deep learning models at a larger scale and make them more accessible to a variety of real world applications. The three following low supervision alternatives for learning deep learning models will be studied in this research work: (1) semi-supervised learning where the labeling task is limited to a small subset of the images but where a large pool of unlabeled samples is available; (2) unsupervised learning where visual concepts are learnt from images only with no need for human annotated labels; (3) automatic annotation utilizing search engines to gather and annotate data using web queries given a set of classes to learn.

More specifically, the quality of the visual features learned by unsupervised learning algorithms has drastically improved in the last 4 years, yet other low supervision tasks only scratch the surface of the synergistic possibilities opened. In most existing cases, unsupervised representations are used in the semi-supervised and label noise literature to either initialize the weights of the CNN to be trained or to train a secondary (regularization) unsupervised objective jointly with the supervised one. The research described in this thesis, will propose to go further and use the visual similarities learned by unsupervised learning algorithms to detect different types of label noise (Chapter 4) or to improve semi-supervised learning using label propagation (Chapter 5).

1.2 Research questions

From the reflections developed in Section 1.1, the following three hypothesis and associated research questions (RQ) emerge.

Hypothesis: Out-of-distribution noise is the dominant noise type in web-crawled datasets. Instead of discarding out-of-distribution samples, they could be used to 1) learn generalizable low-level features to improve the classification accuracy for the in-distribution data, or 2) be used to improve network calibration by promoting under-confident (high-entropy) predictions on out-of-distribution samples at test time.

RQ1: What is the nature of web noise and can detected noisy images be included in the training objective?

Hypothesis: Unsupervised learning has been demonstrated to be a powerful initialisation or regularisation strategy for neural networks when learning image classification tasks. In the case of web-noise, the features learned by unsupervised algorithms contain visual similarity knowledge that could allow the detection of in- and out-of-distribution samples.

RQ2: Can unsupervised learning be used to detect noise in web-crawled datasets?

Hypothesis: Unsupervised learning is a strong strategy when used to initialize weights or as an regularization trained jointly to a semi-supervised classification objective. The similarities learned in an unsupervised manner can be used to discover unlabeled images highly similar to the labeled base.

RQ3: Can unsupervised features be used as a medium to propagate labels in a semi-supervised scenario when few labels are available?

The remaining hypotheses and research questions are related to applications of the above in a real world scenario. It is not evident that low supervision improvements observed of curated benchmarks datasets where labels are discarded will generalize to real world data. More experiments should be conducted in real world situations where labels are very difficult to acquire and where images will not be curated. The last two chapters of this thesis propose to design low supervision solutions to predict herbage composition from RBG images as part of my work in the VistaMilk SFI centre [1]. The following hypotheses and research questions are proposed to study the application of low supervision approaches to real world data.

Hypothesis: Unsupervised and semi-supervised learning have been shown to greatly reduce the amount of supervision needed to perform image classification on curated datasets. These improvements will translate to fine grained real world datasets where ground-truth collection is expensive to collect.

RQ4: Can semi-supervised and unsupervised strategies be devised on specialist, fine-grained datasets such as grass density and composition estimation?

Hypothesis: To be useful to farmers, grass composition prediction needs to be performed on the large areas covered by farms. Drone images offer a scalable solution but renders the collection of ground-truth very time consuming. Semi-supervised strategies are necessary to be able to devise a deep learning algorithm that can infer grass composition and weight from drone images.

RQ5: Can super-resolution and semi-supervised learning be applied to generalise a grass composition prediction model learned on ground-level images to drone data?

[1] vistamilk.ie

1.3 Structure of the thesis

This thesis will begin by presenting algorithmic solutions to train CNNs on web-crawled datasets. Chapter 3 conducts an exploratory study on the type of label noise to expect in web crawled datasets and suggests why label correction approaches, which perform well on synthetically corrupted data, struggle to generalize to web noise. A simple algorithm is then proposed to detect and correct different types of label noise encountered in image classification. Chapter 4 observes that unsupervised contrastive learning can be used to linearly separate in-distribution and out-of-distribution label noise in web crawled datasets and proposes a more complex label noise robust algorithm, utilizing out-of-distribution images to learn low level features in CNNs in a supervised contrastive objective. Continuing with the study of low supervision alternatives for image classification, this thesis will then study semi-supervised solutions for computer vision with Chapter 5 proposing to use label propagation and unsupervised learning to automatically label additional samples to improve semi-supervised learning for image classification. The last two chapters propose semi-supervised research applied to real world applications. Chapter 6 studies how synthetically generated images can be used to reduce the need for human annotations when predicting herbage characteristics and Chapter 7 proposes to use super resolution and semi-supervised learning to perform domain adaptation from herbage images captured on the ground to drone data. Chapter 8 will answer the research questions and conclude the report.

Chapter 2

Literature review

This chapter gives an overview of the relevant literature and publicly available datasets on limited supervision for image classification research. Section 2.1 introduces the curated and low supervision image classification datasets studied in this thesis. Section 2.2 introduces deep learning architectures for computer vision applications. Section 2.3 presents existing strategies to perform unsupervised representation learning on images when no human annotated labels are available. Section 2.4 introduces the state-of-the-art strategies to perform semi-supervised learning for image classification and regression tasks. Section 2.5 introduces the label noise problem and some of the state-of-the-art approaches to train neural networks robustly on corrupted image datasets. Section 2.6 introduces network calibration which will be studied in the case of label noise training in Chapter 3. In the scope of the applied part of this PhD, Section 2.7 will introduce domain adaptation solutions for semantic segmentation algorithms, which will be applied to herbage canopy segmentation using synthetic images in Chapter 6. Finally, section 2.8 presents computer vision solutions for agriculture problems. Some limitations will additionally be given for current state-of-the-art research in semi-supervised and label noise in image classification tasks. Each chapter will include a motivations section that will place the research conducted in the chapter with regards to these state-of-the-art limitations.

2.1 Image classification datasets

Image classification is a prediction task that aims at predicting the category an image should belong to. The quality of an image classification algorithms is often evaluated by its classification accuracy on images unseen during training. An image dataset can be created with more or less amounts of human intervention.

2.1.1 Human curated image classification datasets.

In order to train accurate image classification algorithms, training datasets are usually curated by human annotators. Curated here means that each image in the dataset is presented to multiple human annotators to ensure its relevance to the category it has been assigned to. The most commonly used curated image classification dataset is most probably ImageNet (ILSVRC2012) [100], composed of a million images of mixed resolution (commonly trained on patches of 224×224 px) and divided into $1,000$ classes where each class is composed of $1,000$ image examples. The dataset was created by gathering images from the web and having multiple human annotators curate the data to ensure quality. Since its release in 2012, ImageNet has been an extremely important baseline to compare the classification accuracy of different neural network architectures and training algorithms. Due to the large size of Ima-

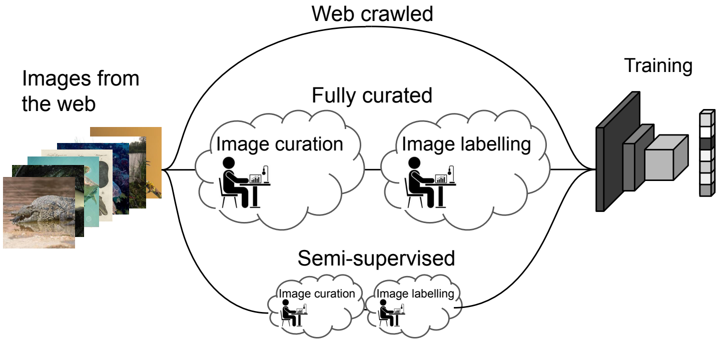

Figure 2.1: Gathering and curating image datasets from the web. Once the images have been retrieved, they can be used directly to train a CNN using the web queries as labels (web crawled). Other options include completely curating the dataset by ensuring that the assigned class correctly describes the object in the image (fully curated) or performing the curation only for a small part of the images (semi-supervised).

geNet, training neural networks on this dataset is computationally expensive. To promote research from laboratories with more constrained resources, some datasets of a reduced size and resolution are often used to reduce experiment times and perform exploratory studies. The CIFAR datasets [99], composed of $60,000$ images of resolution 32×32 divided in 10 or 100 classes are some of the most used low resolution datasets. CIFAR10 and CIFAR100 were created using a similar process as ImageNet [100] using human annotators to curate images gathered from the web. Subsets of the full ImageNet dataset are also available such as miniImageNet [202] composed of 84×84 images divided in 100 classes (600 images per class), tinyImageNet [103] that contains 64×64 images divided in 200 classes (600 images per class), or ImageNet-32(64) [37], which proposes to train the complete but downscaled ImageNet (32×32 or 64×64). Fine grained classification tasks also have associated standardized datasets such as the StanfordCars [98] dataset, which contains $16,000$ images belonging to 196 different car classes.

2.1.2 Limited supervision classification.

Since curating large image datasets using human annotators is a long and expensive task, datasets limiting human intervention have been proposed to the research community. The datasets in the scope of the research presented in this thesis include semi-supervised datasets where only a part of the dataset is annotated by humans such as STL-10 [38] (96×96 images with 10 classes) where in addition to the 100 human-labeled images per class, $100,000$ uncurated unlabeled images are provided to perform semi-supervised learning on. Another type of low supervision datasets are web-crawled datasets, where the human curating and annotation process is omitted. These datasets are directly created using search engines to recover example images for a given category. Figure 2.1 illustrates the different curating options when gathering image datasets from the web for image classification. Because no human curation is involved, these datasets are simple and fast to create but will contain incorrectly assigned images. WebVision [115] is a web-crawled dataset that was constructed on the same classes as ImageNet [100]. The images were gathered from images.google.com (1.1M images) and flickr.com (1.6M images) using ImageNet synsets. To estimate the quality of the dataset, 3 human annotators were assigned the task of voting whether images in a random subset of 200 images per class were inliners or outliers to the class they were assigned to. The authors found that 34% of images were incorrectly assigned (2 or more annotators found the image to be an outlier). The difficulty of training a classification algorithm on WebVision lies in the imbalance of example images per class and the disparity of the noise, as some classes contain much more outliers than others. Clothing1M [213] is another dataset that was crawled from clothes image databases (amazon.com, ebay.com, taobao.com).

Clothing1M contains 1M images split into 14 clothing classes. Because Clothing1M was crawled exclusively from clothes databases, the noisy images are not as diverse as WebVision (images from the whole web) yet the dataset still contains large amounts of miss-assigned images. A similar study as the one conducted on WebVision using human annotators showed that around 40% of the images are assigned to the incorrect category with some classes being much noisier than others. Finer grained web-crawled classification datasets have also been recently proposed to extend label noise algorithms to more complex challenges. Web-aircraft, Web-bird, and Web-cars [183] contain $100, 200, 196$ classes and $13.500, 18.400, 21.450$ images, respectively, and WebiNat-5089 [183] contains 1.1M images in 5089 fine grained categories (plants, insects, reptiles, $\dots$). These datasets were gathered using the Bing Image Search Engine (BISE). Datasets that are an order of magnitude larger have also been proposed, such as YFCC100M [192], a collection of 100M images crawled from flickr.com where the media (image or video) is associated with the metadata of the user which posted the content (title, tags). JFT-300M [182] contains 300M images paired with labels gathered and curated using a mixture of web signals from google.com. LAION-400M [168] contains 400M image-text pairs crawled from the common crawl [1] database between 2014 and 2021 and curated by a trained CLIP [155] model where text-image pairs with a CLIP-predicted cosine similarity under 0.3 were dropped. IG-1B consists of 940M images gathered from instagram.com using hashtags matching with the ImageNet classes. Although YFCC100M and LAION-400M are publicly available, both JFT-300M and IG-1B are proprietary.

[1] commoncrawl.org

2.1.3 Synthetically corrupted datasets

Although the web-crawled datasets presented in the previous paragraph allow researchers to test noise robust algorithms in a real world setting, the noisy nature of the samples in these datasets is unknown. Synthetically corrupted datasets aim to provide a working tool to evaluate how a noise robust algorithm performs on specific parts (clean or noisy) of the dataset during the design phase. A synthetically corrupted dataset is a curated dataset where some of the labels are artificially hidden to the learning algorithm (semi-supervised learning and unsupervised learning) or where noise is methodically injected (label noise). For semi-supervised learning, classification algorithms are typically trained on CIFAR10/100 or ImageNet where only few samples per class keep their original annotation and the rest of the data is considered unlabeled [2, 9, 17]. The amount of remaining data can be as high as 400 per class in earlier semi-supervised works [157, 131, 190] but has been greatly reduced to as few as 4 examples per class in more recent contributions [225]. Introducing synthetic label noise in curated datasets can be performed in a variety of manners. The most basic approach is the symmetric corruption that involves changing the label of a fixed subset of the dataset to a random label from the class pool [8, 150, 159]. Asymetric noise corruption proposes a more realistic setting where the random label assignment of the symmetric corruption is replaced by a semantically close class [145]. For example, the usual label corrupting strategy on CIFAR10 is truck $\rightarrow$ automobile, bird $\rightarrow$ airplane, deer $\rightarrow$ horse, cat $\rightarrow$ dog. Another corruption strategy involves introducing images from different datasets (out-of-distribution samples) [4, 146, 165, 216]. Out-of-distribution in this scope characterises the true label of the incorrectly labeled image. For example in

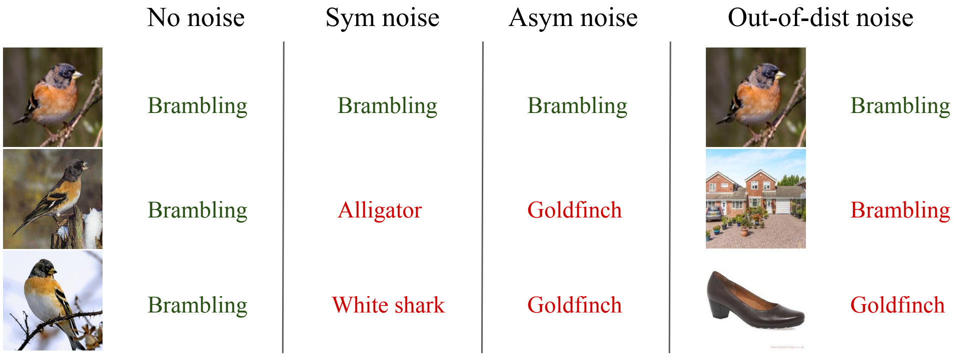

Figure 2.2: Different synthetic noise types encountered in label noise research. Images from WebVision

Figure 2.2, the true label of the image at the bottom right is probably a type of shoe which is outside of the {Brambling, Goldfinch, Alligator, White shark} label distribution i.e. the object in the noisy image does not correspond to any available label in the target classification task. Out-of-distribution images are usually injected into CIFAR10/100 from ImageNet32 [37] or Places365 [230]. Finally, curated datasets can be corrupted using web noise. Jiang et al. [88] tasked human annotators to identify the outliers in images gathered using text-to-image and image-to-image search engines, which recovered examples for classes of miniImageNet [202] and StanfordCars [98] on Google. The identified noisy images recovered can then be used to corrupt a given part of the datasets in a realistic but controlled fashion. Figure 2.2 illustrates different synthetic noise corruption for image classification with 4 classes: brambling, alligator, white shark, and goldfinch.

2.2 Deep learning for Computer Vision

Deep learning architectures for computer vision have become the standard for image classification since the AlexNet [101] architecture was evaluated on the ImageNet dataset [100], outperforming other approaches based on hand crafted features by more than 10 accuracy points. Since then many neural network approaches have proposed. The most notable are: VGG [173], which is an extension of AlexNet, Inception [184], which combines features learned at multiple scales; ResNet [92], which uses skip connections to reduce the vanishing gradients problem, and EfficientNet [187], a suite of meta learned architectures of different sizes. All of these architectures are based on convolutional filters. In 2017, the transformer architecture [200] was proposed. Contrary to the previous networks, the transformer architecture is not convolutional. The transformer was initially proposed for sequential data such as text with Natural Language Processing (NLP) applications using the concept of attention. To summarize the transformer architecture, attention blocks are used to extract a larger amount of context from the input whole sequence. A straightforward example is text translation where a word by word solution would yield limited results. In transformers, attention vectors are computed over the whole input sequence (sentence or paragraph) to compute a contextual embedding that accounts for high attention words to produce the prediction. In the case of language translation, having access to other words in the input sentence will help translate gender or plural rules. Transformers have also been generalized to images [48] by treating the input image as a series of adjacent pixel patches. Although vision transformers reach high accuracy numbers, they require large amounts of data to train their large amount of parameters. The research community has been working on addressing this issue [70, 121]. Although transformers demonstrated accuracy performances superior to convolutional networks for classification tasks, Liu et al. [122] showed that CNNs are still competitive with vision transfomers when the training improvements proposed in the last 10 years by the CNN research community are properly combined (activation function, optimizer, data augmentation).

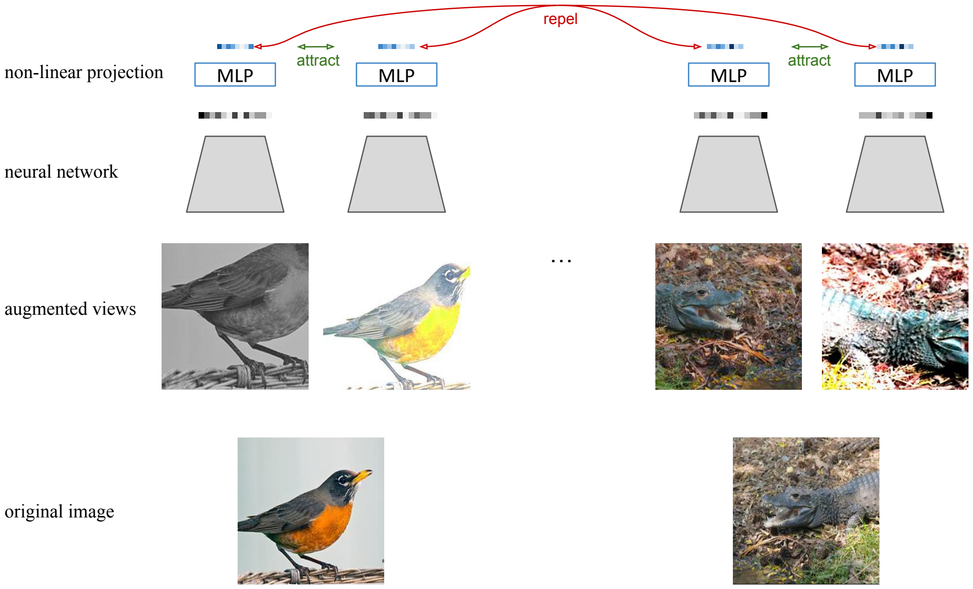

Figure 2.3: Principles of contrastive learning. The features extracted from augmented views of the same image are encouraged to be similar to each other and different from other images. Example inspired by `github.com/google-research/simclr`

2.3 Unsupervised Learning

Unsupervised learning aims to learn features from data alone, independently from human labeling. Early unsupervised learning algorithms for computer vision are mostly self-supervised where the algorithm's goal is to generate its own labels to train on, independently of manual human labeling. Self-supervised tasks include solving jigsaw puzzles [141] where image patches have to be properly ordered, image coloring [229], rotation angle prediction [61] or iterative k-means clustering [25]. In more recent works, contrastive learning has dominated the landscape [30, 72, 206]. The learning paradigm of contrastive learning is to enforce a neural network to learn similar features for different data augmented views of a same image (positive). To avoid the network collapsing to a trivial solution, features extracted from different images in the dataset are encouraged to have low feature similarities (negatives). Figure 2.3 displays a visualization of the principles of contrastive learning. Initial contrastive learning works identified that large batch-sizes were an important factor when training neural networks [30] but latter works proposed that having good negative examples was more important [162]. The most recent unsupervised learning algorithms are instance-based approaches that do not require the negative samples of contrastive learning to compute the loss and use dual networks architectures. In this case, the covariance matrix of the features extracted from two different augmented views of an image passed through two different neural networks is encouraged to be the identity. Some earlier approaches instead minimize the cosine distance between the two representations. In order to avoid collapsing to a trivial solution, different strategies have been proposed. BYOL [65] uses a moving average of the first network to extract the second image view. SimSiam [32] utilizes a stop-gradient to only update one network at a time. Barlow Twins [222] enforces features to be learnt in a non redundant manner by driving non diagonal terms of the covariance matrix to zero. VICReg [14] additionally enforces high variance with features extracted from other images in the mini-batch. Some unsupervised algorithms also propose that similarities and dissimilarities not only be learned between an augmented view and random images from the dataset but that the nearest neighbors be also encouraged to be similar. This includes treating the K nearest neighbors as positives [52] or dynamic confidence-based approaches [56]. Unsupervised learning has also been successfully applied to transformer architectures. DINO [27], which is based on the BYOL [65] algorithm for CNNs, uses a student-teacher transformer architecture where the teacher network is a exponential moving average (EMA) on the weights of the student network. The student is encouraged to learn similar feature embeddings as the teacher when presented with different augmented views of the same image.

2.4 Semi-supervised learning

Semi-supervised learning is a branch of machine learning where only a part of the dataset is annotated while the rest of the data remains annotation free (unlabeled). The relevance of designing semi-supervised algorithms to train neural networks comes from the high amount of human time dedicated to annotating datasets. Semi-supervised learning algorithms devise learning strategies to improve over the performance that could be obtained by training on the few labeled examples by using large amounts of unlabeled samples.

2.4.1 Semi-supervised learning for classification

There exist two dominant strategies for semi-supervised learning: pseudo-labelling and consistency regularization. Pseudo-labeling [9, 21, 45] directly predicts a (pseudo) label for each unlabeled sample in the unlabeled set using initial knowledge learned on the labeled set. The objective then is to update the weights of the neural networks using the labels of the labeled set and the pseudo-labels computed on the unlabeled set, often using strong data augmentation to avoid overfitting. The pseudo labels are updated every epoch as the network learns better representations. Because of the absence of regularization on the predicted pseudo-labels, the pseudo-labelling strategy is prone to produce a network over-confident on its own incorrect prediction as training progresses. This is because the network is encouraged to fit its own direct predictions on unlabeled samples. This limitation is otherwise known as confirmation bias [9, 116]. Consistency regularization [190, 18] proposes to estimate labels by ensembling predictions made on multiple data augmented views of the unlabeled images. Estimating the label for unlabeled image from multiple data augmented views regularizes the pseudo-label guessing process and makes it harder for the network to overfit to its own prediction, effectively reducing confirmation bias. Later iterations of semi-supervised algorithms proposed to combine both a consistency regularization and pseudo-labelling training objective [178]. This combination showed that pseudo-labeling can perform well in the case where only high confidence pseudo-labels are used to train the neural network. The performance is even higher when the pseudo-label confidence threshold is class dependent [225].

2.4.2 Coupling semi- and unsupervised learning

Research contributions have shown that coupling unsupervised and semi-supervised learning can increase the classification accuracy when few labels are available. Rebuffi et al. [158] use RotNet [61] to initialize the weights of a network before starting the semi-supervised process, S4L [223] trains a supervised loss on the few available samples concurently to an unsupervised objective (RotNet or Examplar self-supervision [47]) on both labeled and unlabeled images, ReMixMatch [19] exploits RotNet [61] together with a semi-supervised algorithm to achieve stability when very few labeled examples are available, and EnAET [207] uses transformation encoding from AET [228] to improve the consistency of predictions on transformed images. Finally, SimCLRv2 [31] proposes to first learn unsupervised features using all samples, fine-tune on the few available labels then perform a self-distillation task which is similar to pseudo-labelling.

2.4.3 Label propagation for semi-supervised learning

Label propagation stems from the image retrieval literature and has been used in the semi-supervised literature to propagate labels from the labeled to unlabeled set of the data [16]. Diffusion [46, 186, 231] constructs a pairwise affinity matrix, relating images to each other using meaningful features before diffusing the affinity values to the entirety of the graph. The diffusion result can be directly used to estimate labels and finetune pre-trained networks in few-shot learning [49] or to define pseudo-labels for semi-supervised learning [81]. Other attempts at using label propagation for semi-supervised learning include dynamically capturing the manifold's structure and regularizing it to form compact clusters that facilitate class separation [93], or to encourage random walks ending in the same class they started from, while penalizing different class endings [79].

2.4.4 Taxonomy

Figure 2.4 displays the proposed taxonomy of semi-supervised learning for image classification presented in this literature review. A first distinction is made between semi-supervised algorithms utilizing unsupervised learning objectives or not. For algorithms that utilize unsupervised learning, a dif-

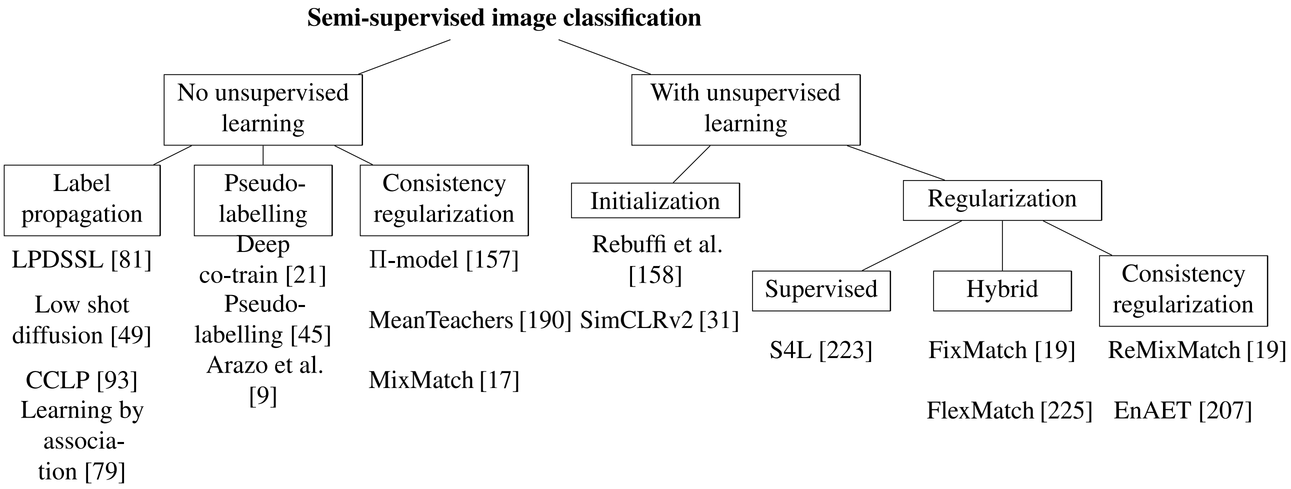

Figure 2.4: Non-exhaustive semi-supervised image classification taxonomy

ferentiation is made between the case where unsupervised learning is used to initialize the network weights or when it is used as a regularization objective to the semi-supervised one. For all algorithms a final distinction is made between pseudo-labelling, label propagation, consistency regularization, supervised or hybrid (consistency regularization and pseudo-labelling) approaches.

2.4.5 Semi-supervised learning for regression

Semi-supervised regression (SSR) solves a regression task on a dataset where the labeled data is limited but the unlabeled data is plentiful. Although semi-supervised classification received many important contributions in the last years, the attention given to SSR has been limited. Timilisina et al. [195] construct a fully connected graph from the feature representations of every sample before performing a bounded heat diffusion process to annotate the unlabeled data. Jean et al. [85] adopt a Bayesian approach by fitting the labeled representations with Gaussian processes and training an auxiliary regularization objective to minimize the predictive variance with regards to the unlabeled points. Bzdok et al. [24] apply an autoencoder on top of medical images of brain voxels to solve a action regression task. The autoencoder is used to compress the input vectors and to ensure that the features extracted from labeled and unlabeled images will be compatible with the final logistic regressor. Li et al. [110] propose a process to aggregate the predictions from multiple regression predictions into a safe pseudo label for the unlabeled samples by means of solving a convex linear combination of each regressor output. Zhou et al. [232] co-train two KNN regressors with different distance metrics that predict pseudo labels to be used by the other regressor on the unlabeled data, effectively reducing confirmation bias. Note that semi-supervised classification algorithms such as consistency regularization approaches [18, 201] or pseudo-labeling [9] should translate to the regression setting. Examples of regression tasks applied to images include age prediction[2] or grass-clover mixture prediction [176].

2.4.6 Limitations of semi-supervised learning for image classification

While impressive improvements have been achieved in terms of classification accuracy improvements together with reducing the number of labeled samples, some problems are still left to be addressed. In all algorithms presented in this literature review, unlabeled examples are considered part of the same curated dataset as the labeled examples where labels have been artificially discarded. This means that all unlabeled examples can be considered as being from the same visual distribution as the labeled ones and that their true label belongs to

[2] https://medium.com/analytics-vidhya/fastai-image-regression-age-prediction-based-on-image-68294d34f2ed

the labeled distribution. In practice, since unlabeled examples should reduce the annotation and data curation cost, they should be considered uncurrated and not all of them will be relevant to the classification task. The STL-10 dataset [38] was proposed to address this real world scenario but too many labeled samples are proposed for the current state-of-the-art standards. To account for the real world scenario of semi-supervised learning, Ren et al. [160] proposed to select only relevant unlabeled samples to be used in the semi-supervised objective to avoid learning from underconfident pseudo-labels to improve convergence and the final classification accuracy. Another question raised by recent semi-supervised algorithms is the importance of data augmentation in the semi-supervised strategies as more and more advanced augmentation strategies such as AugMix [74], CTAugment [19] or the strong SimCLR augmentations [30] are utilized. This begs the question of whether recent improvements are principally due to these strong augmentations instead of the proposed training strategies. Comparing semi-supervised approaches independently of the data augmentation used is further complicated by the use each algorithm makes of the augmentation e.g. to guess better pseudo-labels [178] or to avoid overfitting [19] (confirmation bias). A common code baseline (similar to solo-learn [40] for unsupervised learning) would be desirable to render the comparison of semi-supervised algorithms unbiased to the augmentation strategies. Finally, some efforts are made to couple unsupervised and semi-supervised learning but these are often limited to network pretraining [31, 158] or unsupervised regularization [19, 178], that boost the baseline classification accuracy yet miss the opportunity to use similarities learned in an unsupervised manner to transfer label knowledge from the labeled to the unlabeled samples. Chapter 5 will propose to use the strong representation learning power of recent unsupervised learning algorithms to extend the size of the labeled set using unsupervised affinities between labeled and unlabeled samples.

2.5 Label noise

Label noise research is a rapidly developing area inspired by the use case where the human curation of images and annotations is completely left out when building datasets from the web. Datasets gathered using web queries have the advantage of being easy to assemble and to be already labeled by the query. The main problem in web datasets is that some of the retrieved images will be mismatched with their associated query (outliers). In the scope of label noise robust algorithms, the clean or noisy nature of samples is unknown. Label noise robust algorithms aim at detecting and correcting the incorrectly labeled images in order to improve the generalization of the algorithm. The research conducted in this thesis will not rely on the possible availability of a known clean or noisy set of images.

2.5.1 Robust losses to combat label noise

Loss correction algorithms aim to reduce the contribution of noisy labels in the loss used to train the weights of the neural network. Some naturally robust losses have been proposed where theoretical guaranties ensure that the gradient of the incorrectly labeled images remains small [119, 125, 210, 218]. More specifically, Patrini et al. [150] propose, given that the transition matrix between classes is known, to correct the gradient of the loss using a matrix multiplication with the class noisiness prior to readjust the expected incorrect network predictions. Reed et al. [159] compute a linear interpolation between the loss against the ground-truth label (possibly noisy) and the loss against the network's own pseudo-label (possibly correct) using a fixed hyper-parameter.

2.5.2 Individual detection of noisy samples

A more modern take on label noise robustness is to explicitly detect the noisy samples. Arazo et al. [8] observe that because CNNs are naturally robust to noise, correctly labeled samples that share similarities with other correctly labeled examples from the same class will be learned more easily than their noisy counterpart. The authors show that when observed at the right moment (before overfitting to the noise) the histogram of the per-sample training loss appears bi-modal. The low loss mode contains the clean samples and the high loss mode contains the noisy ones. A Gaussian Mixture is fit to detect the high and low loss modes in the historgram and a correction strategy is applied to the noisy samples where the network current prediction (pseudo-label) is used as the corrected label. MentorMix [88, 87] uses a mentor (or teacher) network, which is an exponential moving average (EMA) of the weights a student network. If the predicted loss of the mentor network is less than a hyper-parameter threshold, a high importance is given to the sample in the training loss. Detected noisy samples above the loss threshold are weighed with values close to 0 to reduce their impact on the student loss and weight updates. Sample representations in the neural network feature space has also been used to detect noisy samples. In MOIT [145] noisy samples are identified as having many neighbors from a different class than its assigned ground-truth.

2.5.3 Correcting detected noisy samples

Multiple alternatives have been proposed in the literature to correct the detected noisy samples. Semi-supervised correction for label noise is a successful approach that proposes to estimate the true label of detected noisy samples using semi-supervised approaches. Ortego et al. [143, 145] apply a pseudo-label based guessing strategy [9] while DivideMix [111] and JoSRC [216] use consistency regularization [17]. In the case of RRL [113] corrected labels for noisy samples are guessed using a weighed average of the 200 closest clean neighbors in the network feature space. Another alternative is to simply discard the detected noisy samples and train the neural network on the detected clean samples only [87, 88, 170, 220]. This approach appears to be especially competitive on real-world web datsets where the noise is mostly out-of-distribution.

2.5.4 Out-of-distribution noise

Recent label noise research has proposed to tackle out-of-distribution noise in addition to in-distribution noise. The true label of out-of-distribution noisy samples lies outside of the label distribution that the neural network is trained to predict. In this case, label correction is impossible. Algorithms trained for classification on datasets where both in-distribution and out-of-distribution noise is present hypothesize that a neural network will behave differently on the two noise types. The softmax predictions of the neural network on out-of-distribution samples is expected to be under-confident while predictions on in-distribution samples should be confident but in a different class than the noisy assignment. JoSRC [216] evaluates the Jensen-

Shannon divergence between the prediction of the CNN on two augmented views of the same image. If the divergence metric is high, i.e. the network outputs different predictions for the same image augmented differently, the prediction is deemed under-confident and the sample is tagged as out-of-distribution. EDM [165] evaluates a metric called the evidential loss on the network predictions. When observing an histogram of the evidential loss values over the whole dataset, three modes can be observed that indicate (from low loss to high loss) the clean samples, out-of-distribution samples, and in-distribution noisy samples. Each subset is recovered using a three mode Gaussian mixture. Other contributions, such as RRL [113], propose to tag a noisy sample as out-of-distribution if the confidence (highest bin in the softmax class probabilities) of the corrected label (guessed using a vote between the labels of the N closest clean samples in the feature space) is inferior to a hyper-parameter threshold.

2.5.5 Regularization

Network regularization using data augmentation such as Mixup [227], a strategy based on image and label interpolation, has been shown in Arazo et al. [8] to be very robust to label noise and is almost systematically used in label noise robust algorithms. Regularization loss terms are also used where the predictions are encouraged to have a low entropy (very confident) or where the predictions of the network are encouraged to be class balanced [111, 119, 188]. Unsupervised regularization is also used where an unsupervised learning objective (insensitive to label noise) is minimized together with the supervised. For example, Li et al. [113] minimize the unsupervised SimCLR [30] objective together with a noise robust supervised objective.

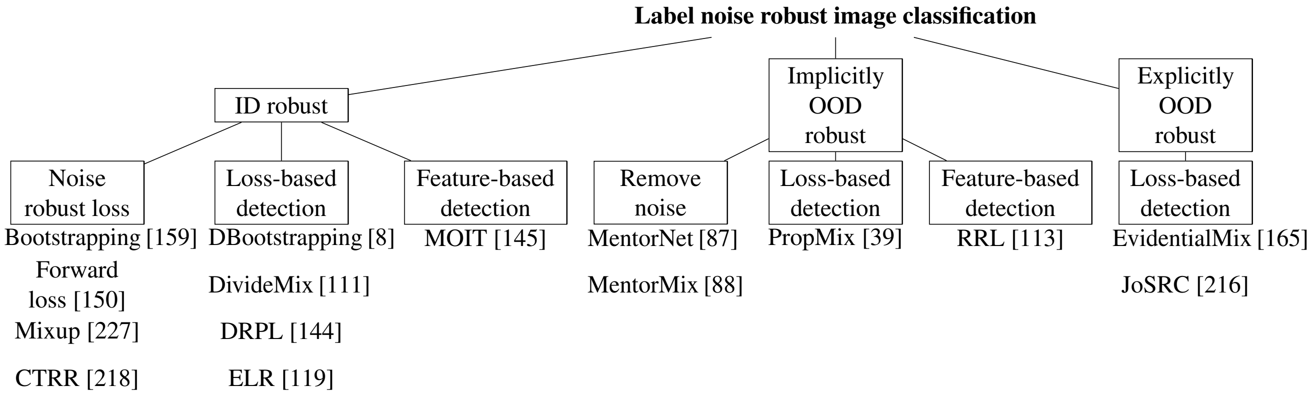

Figure 2.5: Non-exhaustive taxonomy for image classification in the presence of label noise

2.5.6 Taxonomy

Figure 2.5 displays the proposed taxonomy of label noise for image classification presented in this literature review. The principal distinction made between noise robust algorithms is whether they account for out-of-distribution (OOD) noise during the noise detection phase (no OOD detection, implicit OOD detection, explicit OOD detection). Implicit OOD detection refers to approaches that remove all detected (ID and OOD) noisy samples, or that avoid relabelling detected noisy samples in the case where the predicted true label is estimated to be incorrect or insufficiently confident. Algorithms are then differentiated between each other depending whether or not noisy samples are explicitly detected, and if they are, on how noisy samples are detected: using a loss-based approach or feature space representations.

2.5.7 Limitations of the label noise state-of-the-art

Most algorithms presented in section 2.5 are designed using a curated dataset altered artificially to introduce in-distribution label noise where the labels of a fraction of the dataset are randomised. Knowing the nature of the noisy samples is necessary at design time for producing a robust test time algorithm. The direct real world application of label noise robust algorithms are uncurated datasets directly crawled from the web [115, 183, 213]. In these real world datasets, it is fair to estimate that a large portion of the noisy images are out of the desired classification distribution and algorithms designed on curated datasets, synthetically corrupted with in-distribution label noise would not address this issue. Designing algorithms robust to a mixture of in- and out-of-distribution noisy images is desirable. To make it possible to design such algorithms, Jiang et al. [88] proposed a controlled web label noise dataset based on MiniImageNet [202] where the web noisy samples are known. Chapter 3 will propose to conduct a study to evidence the in- or out-of-distribution nature of web noisy samples in the mini-WebVision [115] dataset and a metric to independently retrieve both types of noise. Because unsupervised learning learns representations independently of man-made labels, these unbiased representations should be relevant as part of a noise-robust algorithm to detect labeling errors. Some algorithms have proposed to use unsupervised learning as an unsupervised regularization to a noise-robust supervised objective [113] or as a weight pretraining for the neural network [39]. Similarly to the point made in the limitations of semi-supervised learning for image classification, the visual similarities that can be learned between images in an unsupervised manner have not yet been used in the literature and could be very efficient at identifying out-of-distribution images. Chapter 4 will study how unsupervised contrastive learning can be used to accurately identify both in- and out-of-distribution images in web-noisy datasets by clustering their representations in the unsupervised feature space.

2.6 Network calibration

Network calibration is an important aspect of DNN training that ensures that the probability distribution of the prediction of a network matches the observed ground-truth distribution. This especially implies that neural networks should not be over or under confident so that the confidence of the network in its own predictions can be properly interpreted by a end user. Training a well-calibrated network is highly desirable for high-risk applications such as medical diagnosis [51, 156]. Training on label noise datasets will degrade the calibration performance of a neural network. Expected Calibration Error (ECE) is a widely used metric to evaluate the calibration of a network [66] that penalizes both over-confident and under-confident predictions. Other notable calibration metrics include: Over-confidence Error (OE), which penalizes only over-confident predictions [193] and Maximum Calibration Error (MCE) [66], which penalizes only the maximum difference between prediction and ground-truth for a given class and not the average over all classes. Mixup [227], a data augmentation strategy, has shown benefits for both robust label noise training [8] and network calibration [193]. Chapter 3 will study how out-of-distribution images can help imporve network calibration when training on a label noise dataset.

2.7 Semantic segmentation on synthetic images and domain adaptation

Semantic segmentation aims at predicting the object that each pixel in an image belongs to [55]. The human annotation required for semantic segmentation tasks is extensive, often requiring several hours per image [118]. This

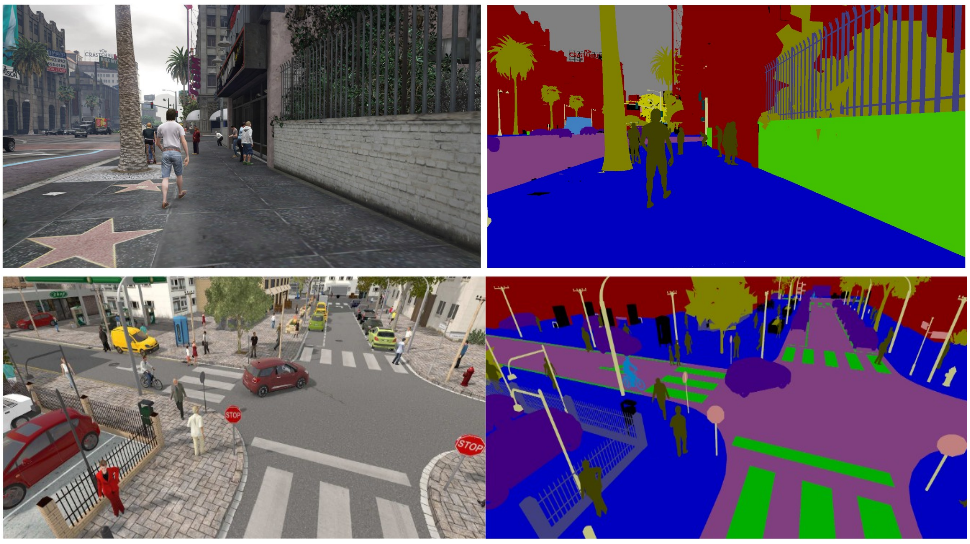

Figure 2.6: Example of synthetic image datasets. Images and semantic segmentation ground-truth generated from the GTA V graphics engine (top) and from the SYNTHIA dataset (bottom)

makes training strategies using fewer human annotated images attractive. Synthetic images promise to solve part of the problem by providing an unlimited amount of perfectly segmented training images. Popular synthetic datasets for semantic segmentation include Grand Theft Auto V (GTA V) [161] or SYNTHIA [163] that create synthetic images of cities using graphics engines. Figure 2.6 displays synthetic image examples from both datasets.

Although the large quantity of labeled data allows a semantic segmentation neural network to converge on a synthetic dataset, the results need to generalize to real world data. Domain adaptation aims at learning domain agnostic features that can generalize from synthetic data to the real world. Domain adaptation strategies can be applied at different stages in a network: input adaptation, feature adaptation, or output adaptation.

Input adaptation strategies aim to transform synthetic images to look more realistic by applying a visual style, often using a Generative Adversarial Network [35, 167, 172, 233].

Feature adaptation approaches aim to discover domain invariant (or aligned) features between synthetic and real data. Chen et al. [29] propose to use a maximum square loss to enforce a linear gradient increase between easier and harder classes. Luo et al. [124] use a significance aware adversarial information bottleneck; Chen et al. [34] propose a knowledge distillation approach by matching network activations to a network pretrained on ImageNet.

Output adaptation techniques constrain the network prediction directly to enforce better generalization. This can be achieved using adversarial approaches where the predictions made on synthetic and real data should be indistinguishable to a discriminator network [20], or by enforcing low entropy (more confident) predictions [203]. Finally, batch normalization fine tuning on real data, where the batch normalization parameters are tuned on the real images before evaluation, has also been shown to be a simple but effective domain transfer strategy [117] even when only a few hundred images are available for the target domain. A further study of domain adaptation for semantic segmentation can be found in the extensive domain review of Toldo et al. [196]

2.8 Computer vision for agriculture

The last two chapters of this thesis propose solutions to perform plant phenotyping and dry matter predictions from RGB images. Agricultural problems are excellent domains for the application of image analysis approaches since computer vision can be used to be extract relevant information from the environment at a large scale and in a non-destructive manner. Existing works explore a variety of computer vision applications for plant detection and identification. This section will review some of the most relevant to this thesis.

A common problem tackled by computer vision in agriculture is to monitor the proper development of crops through weed, pest and disease detection [151, 194]. Weed detection aims to localize unwanted weeds to ultimately remove them, either by hand or using a robot. Existing approaches include employing color filtering, edge detection, and area classification [147, 153, 189]; utilizing color features used to train random forest algorithms and support vector machines [82, 83]; or using neural networks to semantically segment images [102]. Disease detection aims to precisely identify areas of a plant that are infected and deploy localized treatments. Toseef et al. [199] utilize a fuzzy inference system to diagonize diseases in wheat and cotton plans, Wang et al. [205] detect black rot in apples using a small convolutional neural network and Zhai et al. [224] propose to optimize UAV flight paths using genetic algorithms and particle swarms for automatic weed spraying.

Fruit or vegetable detection and counting is also of interest because it has the potential to greatly reduce human labor by enabling automatic fruit treatment or collection on the farm. Examples include tomato segmentation and counting using a convolutional neural network [1], large scale fruit detection in trees [164], or real-time fruit detection using a lightweight neural network [22].

2.8.1 Drone (UAV) imaging

Drones hold important potential for automating farm tasks since drones can easily cover large areas of uneven terrain [211]. Deep learning has been successfully applied to derive growth rate from nitrogen fertilization on drone images [77], estimate the emergence rate of seeds in the field [120], wheat density [90], weed detection [54], land classification [36], and wheat head identification [42]. In addition to RGB imaging, additional sensors such as radars or lidars [109] can be accumulated to improve performance [127]. The main drawback when applying deep learning on drone images for plant phenotyping remains the difficulty of ground-truthing the images because of the large areas covered [211].

2.8.2 Generative adversarial networks (GANs)

GANs are of special interest in the plant domain because their generative properties can allow them to forecast growth [50] or perform domain adaptation [176]. We separate here GAN architectures between the conditional architectures that are trained with corresponding pairs of input and outputs represented in the two different visual domains [84], and unpaired architectures, e.g. CycleGAN [233] or Contrastive Unpaired Translation [148] where images from both visual domains contain different semantics. Conditional GANs have been successfully applied to generate RGB images from semantic segmentation masks [234], to predict cabbage growth [50], or plant super-resolution to improve feature detection [41]. Unpaired GANs have been used to estimate disease spreading on leaves [112, 136, 137], or to improve the realism of synthetic images [15, 69]. Figure 2.7 displays examples of GANs being successfully used to forecast cabbage growth [50] or generate examples of plants affected by powdery mildew [137] .

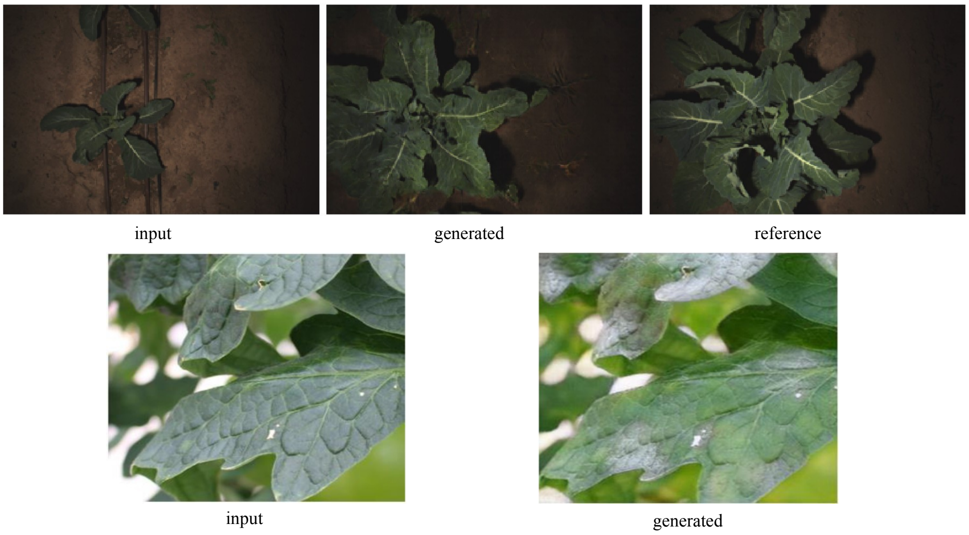

Figure 2.7: Example of GANs applied to agriculture problems. Cabbage growth forecasting(top) and powdery mildew infection (bottom)

.

2.8.3 Biomass composition estimation from canopy view images

Being able to estimate grass composition from canopy images opens up solutions for autonomous targeted fertilization in fields. Automated fertilization reduces costs for the farmer and water pollution due to over-fertilization [6, 177]. The heavy occlusions present in canopy images poses significant challenges as the biomass estimate should account for elements hidden from the canopy view. Himstedt et al. [76] study the biomass of clover in a legume-grass mixture and demonstrate a good capacity to detect clover from the legumes using morphological filtering and color segmentation to detect the clover. The authors were then able to accurately predict the clover biomass in a controlled environment under the assumption that the total biomass is known. Mortensen et al. [132] propose to segment the grass clover mixture using color filtering and edge detection before employing a linear regressor to learn the mapping between coverage area of each species and dry biomass content. The authors were then able to directly predict the dry biomass of each element from the image alone. In the case where the herbage will be stored for winter to feed cows, estimating the dry biomass from an image of the fresh pasture directly becomes of interest. Skovesen et al. [175] propose an improvement over previous work by using a neural network to segment images, and then fit a linear regressor to the detected species percentages to predict the biomass percentages. To train the neural network, a synthetic dataset is generated where grass/clover/weed elements are manually cut from the raw images and pasted in a random fashion on a soil background image to create a synthetic but fully segmented image. This allows the authors to generate an infinite amount of ground-truthed training images from a similar visual domain to train the segmentation algorithm. An updated version of the segmentation algorithm from synthetic data was later published [176] using style transfer GANs to simulate different weather conditions and multi-resolution prediction. Based on this work, the GrassClover dataset challenge [174] asks entrants to improve the author's baseline using the synthetic images together with a large collection of unlabeled real images and a small set of manually labeled real images. The Irish grass clover dataset [75] proposes, additionally, to the dry biomass percentages, to predict the herbage height pre-grazing (cm) and the dry matter per hectare (kg DM/ha) from the canopy images. Although both datasets provide an additional large amount of unlabeled images, the respectively baseline are purely supervised and do not make use of the raw images. Subsequent algorithms were published and tested on both datasets to attempt to use the raw data to improve the biomass prediction. Narayannan et al. [135] proposed to use mean imputation to infer labels for the partially labeled samples before training a convolutional neural network (CNN) on the larger dataset. Albert et al. [5] instead uses an unsupervised learning algorithm [107] on the unlabeled data to learn better initial representations that allow for better accuracy numbers with limited amounts of labels.

Chapter 3

Addressing out-of-distribution label noise in webly-labelled data

This chapter presents a novel approach to deal with label noise inherent to datasets created using web queries. Our approach is simpler than previous works yet sets a new state-of-the-art for the Webvison1.0 dataset. Section 3.1 motivates automatic gathering of datasets using web queries; the limitations of existing approaches for dealing with the specific kind of label noise caused by the automatic annotation process; an introduction to the proposed method and the contributions of the chapter. Section 3.2 presents an exploratory study of the WebVision 1.0 dataset where we look at the type of label noise present in the dataset. Section 3.3 formalizes our algorithm. Section 3.4 presents implementation details, an ablation study for the different elements of our method, and a comparison with the state-of-the-art. Section 3.5 concludes on the findings and the results of the proposed method. The research that emanated from this work was published at the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV).

3.1 Motivations

As discused in Chapter 2, deep Neural Networks (DNNs) are now the standard approach for accurately solving image classification tasks [187, 221]. However, their principal drawback is the large amount of labeled examples required for training. There exist numerous alternatives to deal with the limited availability of labels, such as but not limited to, semi-supervised learning [2, 9, 18], unsupervised learning [61, 30] and robust training on automatically annotated datasets [115, 88]. This chapter focuses on the latter.

Designing robust algorithms to train image classification DNNs in the presence of label noise is an important focus for the community [181]; these enable better adaptation of current DNN solutions to real-world problems where extensive curated datasets are unavailable or too expensive to build. In order to design and evaluate noise-robust algorithms it is common to create controlled label noise datasets by synthetically introducing label corruptions in traditional comparison benchmarks such as CIFAR10/100 [99]. Although good noise robustness is shown on these artificial datasets, web label noise has proven more challenging because these solutions generalize poorly to this more realistic scenario and can, in specific cases, be outperformed by a non noise-robust strategy trained with robust data augmentations such as Mixup [144, 88].

We hypothesize that the main limitation for the correction of label noise in web crawled datasets comes from a common assumption made by most label noise robust algorithms [159, 150, 111, 204] where the true labels for noisy samples lie inside the label set, i.e. the label noise is *in-distribution* (ID). Conversely, we hypothesize that the label noise present in web crawled datasets is predominantly *out-of-distribution* (OOD) meaning that the true

Research published in the Winter conference on Applications of Computer Vision (WACV) 2022 label of the images lies outside the label pool and the real labels for OOD noisy samples can not be corrected to another from the label distribution. To confirm our hypothesis, we conduct a small but representative survey on the WebVision 1.0 dataset [115] to identify the type of noise one can expect in automatically annotated datasets crawled from the web. We then make use of controlled open sourced web label noise datasets provided by [88] to build and validate a simple method which separately detects and corrects ID and OOD noise. We argue that training a well-calibrated network [66, 140] is an essential aspect of label noise robustness, especially in the presence of noisy OOD samples, as a well-calibrated network avoids over-fitting OOD noisy samples by reliably predicting under-confidently on these outliers. To this end, we choose to use the out-of-distribution noisy samples to improve network calibration rather than simply discarding them.

3.2 Exploratory analysis of web datasets

Recent state-of-the-art for label noise detection and correction relies on strong assumptions verified on synthetically generated noise. [144, 88] demonstrated that many algorithms developed on synthetic datasets do not generalize well to real-world label noise and that improvements are often inferior to using data augmentation. We hypothesize that this limitation is a consequence of noisy samples having their labels corrected by assigning another label from the known label distribution, i.e. the noise is in-distribution. We conversely hypothesise that most of the noise in web labeled datasets is out-of-distribution, and hence the real unknown label lies outside of the known label set. To verify this hypothesis we randomly sample images from the real-world la-

Table 3.1: Analysis on the noise types and ratios found in mini-WebVision. We randomly sample three subsets (S) of 2000 images and report correctly-labeled samples and in-distribution (ID) and out-of-distribution (OOD) noisy samples.

	S1	S2	S3	Average (%)
Correct	1441	1440	1335	1405.33 (70.30)
OOD	460	429	573	487.33 (24.38)
ID	98	130	91	106.33 (5.32)

bel noise dataset mini-WebVision (first 50 classes subset of the WebVision 1.0 dataset [115]) and manually categorize their label in three categories: clean, in-distribution noise, and out-of-distribution noise. Table 3.1 shows the results of the study, demonstrating that there is a clear domination of out-of-distribution noise over in-distribution noise. This observation sheds light on the limited improvements of in-distribution label correction techniques when applied to web crawled datasets, while explaining the benefits of algorithms that sample noisy data less often to reduce their contribution [67, 88]. The annotations used are available in our code repository.

3.3 DSOS

Taking into consideration the results observed in Section 3.2, we propose Dynamic Softening of Out-of-distribution Samples (DSOS), a label correction algorithm for robust learning on web label noise distributions. We aim to solve an image classification task over C classes as learning a DNN model h given a training set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ of N samples where $x_i \in \mathcal{X}$. More specifically, we tackle the case where the dataset consists of a correctly labeled set $\mathcal{D}_c = \{(x_i, y_i)\}_{i=1}^{N_c}$ with corresponding one-hot encoded labels $y_i \in \{0, 1\}^C$, an incorrectly labeled in-distribution noisy set $\mathcal{D}_{in} = \{(x_i, y_i)\}_{i=1}^{N_{in}}$

and of an out-of-distribution noisy set $\mathcal{D}_{out} = \{(x_i, y_i)\}_{i=1}^{N_{out}}$. We denote $N = N_c + N_{in} + N_{out}$ the total number of available samples. We consider unknown the distribution of the samples between $\mathcal{D}_c$, $\mathcal{D}_{in}$ and $\mathcal{D}_{out}$. We note $h : \mathcal{X} \rightarrow [0, 1]^C$ the deep neural network (DNN) we train to classify the images as belonging to a class $c \in \{1, \ldots, C\}$.

3.3.1 Separate detection of ID and OOD noise

Motivation

We motivate here the need for a new metric for the dual detection of ID and OOD noise in web crawled datasets by considering the ideal case where a network has been trained on a web-crawled dataset and did not overfit the noise. Samples would then be characterized by either a confident correct prediction (clean samples), a confident incorrect prediction (ID noise), or an un-confident prediction (OOD noise). Noise detection metrics from in the label noise literature propose to either quantify the accuracy of the prediction [8, 111, 68] (cross-entropy loss, accuracy, Kullback-Leibler divergence) or the uncertainty of the prediction [198, 216] (forgetting events, entropy of the prediction, contrastive predictions). Relying on one characterization of the network prediction alone is problematic when presented with the duality of the noise present in web-crawled datasets as ID and OOD noise cannot be independently retrieved. While accuracy approaches indistinguishably retrieve incorrectly predicted OOD and ID noise (both having low agreement with their noisy label), certainty-based approaches only retrieve under-confident OOD noise. EvidentialMix [165] proposes an independent retrieval of ID and OOD noise, where a mean square error + variance loss [169] (evidential loss,

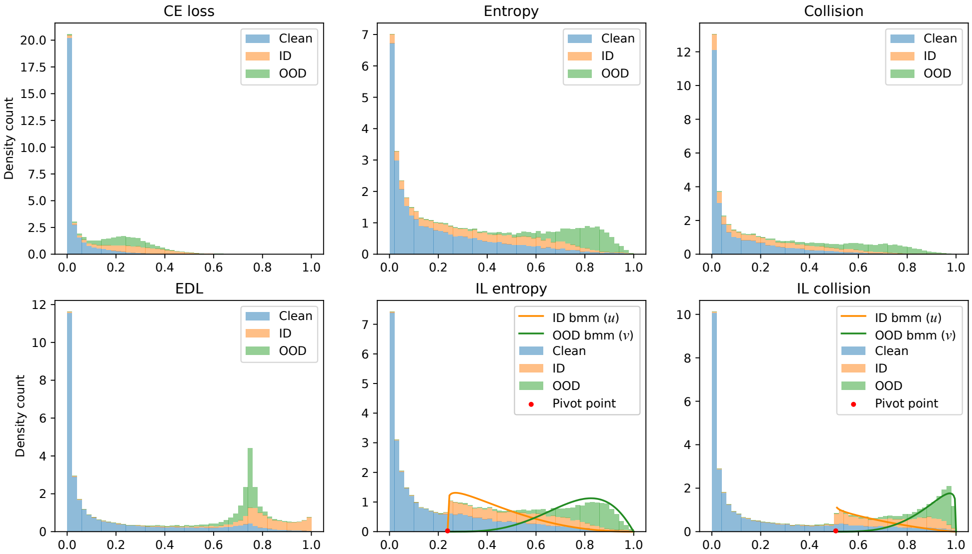

Figure 3.1: Stacked density histograms for multiple noisy sample retrieval measures on CIFAR-100 with $\rho = \psi = 0.2$. All metrics are min-max normalized. For the entropy of the interpolated label (IL) we also draw the decision function (BMM) that we fit to the data. The pivot point in red separates clean from noisy samples.

EDL) is shown to separate ID and OOD noise on artificial corrupted noisy datasets (CIFAR-10 [99]. We argue that the limitation of the evidential loss for web-crawled datasets lies in the absence of separation between OOD noise and lower-confidence predictions in general, resulting in a sub-optimal OOD retrieval, the dominant noise type for web-crawled datasets. This limitation is evidenced in Figure 3.1 and in Table 3.2 where we compare retrieval scores for Clean/ID/OOD samples (one versus all) for an accuracy (CE loss) or confidence metric (entropy) against using the EDL loss fitted with a 3 components Gaussian mixture model [165], and two variations of our proposed metric. Table 3.2 highlights the trade-off we make for better OOD detection at the cost of less accurate ID retrieval when compared with EDL and further discussed in Section 3.3.1.

Table 3.2: AUC retrieval score for different types of metrics after warm-up on CIFAR-100 with $\rho = \psi = 0.2$. Higher is better.

	Clean	ID	OOD
Small loss	95	87	81
EDL	93	90	75
IL entropy	91	81	94
IL collision	93	85	92

Dual noise detection metric

We propose a novel noise detection metric that allows the separate detection of confident clean samples, confident ID noisy samples, and OOD noisy samples. To do so, we propose to compute the interpolated label between the current network prediction $\hat{Y}$ and the target label Y: $y_{int} = \frac{y_i + \hat{y_i}}{2}$ and to study its collision entropy:

$$l_{detect} = -\log\left(\sum_{c=1}^{C} y_{int,c}^2\right). \tag{3.1}$$

We aim to detect three different events for y_{int}: the clean event where prediction and ground truth agree, resulting in a low entropy; the ID event where prediction and ground truth are both confident but disagree (medium entropy); and the OOD event where the prediction is under-confident (high entropy). A visualization of these events is available in Section 3.4.6. Studying the entropy of the interpolated label l_{detect} allows us to reverse the detection hierarchy observed in the EDL from clean-OOD-ID to clean-ID-OOD since confident incorrect predictions are now observed in y_{int} as a bimodal distribution that has a lower entropy than an interpolation of the ground truth with an un-confident uniform prediction. A fundamental property of l_{detect} is that it differentiates between low confidence but correct predictions (clean samples) and confident incorrect predictions (ID noise), which is evidenced by the pivot point. The pivot point is defined for y_{int} being a perfect bi-modal distribution, i.e. two high probability modes with values 0.5 with all other bins to 0, resulting in $l_{detect} = -\log 0.5$, the pivot point. Detecting these events of high probability motivate our choice of using the collision entropy, which is more sensitive to high probability events than the Shannon entropy. Using the pivot point together with the observed bimodality of the noisy samples, we classify the samples in three distinct categories where every sample whose l_{detect} value is inferior to the pivot point is considered clean and where we fit a two components Beta Mixture Model (BMM) to the noisy samples. By computing the posterior probability of a sample to belong to each component, we evaluate the ID and OOD nature of every noisy sample.

Figure 3.1 illustrates the clean/ID/OOD separation observed for accuracy based and uncertainty based metrics on the CIFAR-100 dataset corrupted with 20% symetric ID noise and 20% OOD noise from ImageNet32 [37] at the end of the warm-up phase (see Section 3.4.1 for training details). The figure illustrates how the collision entropy improves the separation between clean and ID noise over the Shannon entropy and how we trade off improved OOD detection for a decreased ID detection over the evidential loss (EDL) [165] (see Table 3.2). The pivot point is indicated in red. An additional illustration explaining the behavior of l_{detect} for intermediate configurations of y_{int} is available in Section 3.4.6.

3.3.2 DSOS

We build DSOS as a single network based, single training cycle algorithm which aims to first discover ID and OOD samples in a corrupted dataset before separately addressing ID and OOD noise using dynamic label correction

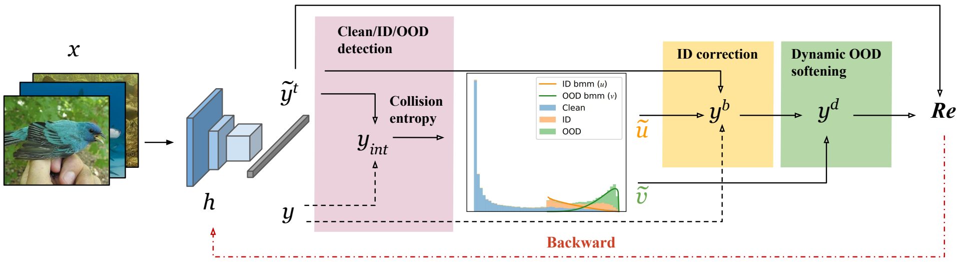

Figure 3.2: Visualization of the DSOS algorithm. DSOS identifies and corrects the ID and OOD noise from the training distribution before applying targeted label correction.

strategies. Figure 3.2 illustrates the DSOS algorithm. We aim to correct ID samples using confident predicted label assignments and to encourage high entropy prediction for OOD samples which cannot be corrected. DSOS aims to minimize the following empirical risk over the noisy dataset:

$$R_e = \frac{1}{N} \sum_{i=1}^{N} -{y_i^t}^T \log h(x_i), \tag{3.2}$$

where the logarithm is applied element-wise and y_i^t denotes the, possibly unknown, true label for sample x_i. Although it is possible to directly minimize R_e for ID noisy samples by correcting the noisy label y_i to the true label y_i^t, this is not the case for OOD label noise. We propose then not to attempt to approximate the true label of OOD samples using a label from the known distribution but instead to promote better network calibration by encouraging high-entropy predictions, i.e. a uniform prediction over ID classes. We then rewrite empirical risk as:

$$\begin{aligned} R_e = &- \frac{1}{N_c + N_{in}} \sum_{i=1}^{N_c+N_{in}} {y_i^t}^T \log h(x_i) \\ &- \frac{1}{N_{out}} \sum_{j=1}^{N_{out}} {y_s}^T \log h(x_j), \end{aligned} \tag{3.3}$$

where y_s is the softened label, i.e. a perfect uniform prediction over all the classes C. To obtain a dynamic softening from y_i^t to y_s and given a OOD classifier $\mathcal{V} = \{v_i\}_{i=1}^N, v_i \in [0, 1]$ where $v_i = 0$ means sample x_i is OOD, we minimize:

$$R_e = -\frac{1}{N}\sum_{i=1}^{N} f(y_i^t, v_i)^T \log h(x_i), \tag{3.4}$$

with $f(y_i^t, v_i)$ the smoothing function where $f(y_i^t, 0) = y_s$ and $f(y_i^t, 1) = y_i^t$.

Label softening of out-of-distribution samples

We minimize the risk in Eq. 3.4 using a label correction approach where we aim to first correct the labels for noisy ID samples to their true label using a bootstrapping inspired approach [8, 159, 180]. For the OOD samples, we propose a dynamic softening strategy by computing the cross-entropy loss with regards to a dynamically smoothed label (the more likely a sample is detected to be OOD, the more uniform the target) and avoid using an additional regularization term (Kullback-Leibler divergence minimization between the prediction and a uniform target would be a common solution [108]). To correct ID label noise, we consider a first estimated metric $\tilde{U} = \{\tilde{u}_i\}_{i=0}^N$, where $\tilde{u}_i \in \{0, 1\}$, evaluating whether a sample is noisy but in-distribution, i.e. the label can be corrected to another from the distribution. $\tilde{u}_i = 1$ denotes sample x_i is noisy but ID. We denote $\hat{y}_i^t$ the current true label guess for sample x_i an correct it with,

$$y_i^b = (1 - \tilde{u}_i)y_i + \tilde{u}_i\hat{y}_i^t. \tag{3.5}$$

Regarding OOD label noise, we consider a second metric $\tilde{V} = \{\tilde{v}_i\}_{i=0}^{N}$ estimating $\mathcal{V}$ and evaluating whether a sample is noisy and OOD ($\tilde{v}_i \in (0, 1]$) with $\tilde{v}_i = 0$ meaning a sample is considered OOD. We re-normalize the possibly bootstrapped label y_i^b for a sample x_i assigned to an OOD noisiness metric estimation $\tilde{v}_i$ as

$$y_i^d = \frac{\exp\left(\frac{\tilde{v}_i y_i^b}{\alpha}\right)}{\sum_{c=1}^{C} \exp\left(\frac{\tilde{v}_i y_{i,c}^b}{\alpha}\right)}. \tag{3.6}$$

with $\alpha \in [0, 1]$ a hyperparameter. y_i^d is a dynamically smoothed correction of the corrected label y_i^b where $\frac{\tilde{v}_i}{\alpha}$ serves as a dynamic temperature depending on the out-of-distribution noisiness of the sample. In Figure 3.1, $\tilde{U}$ corresponds to the posterior probability given l_{detect} for the left-most beta mixture being superior to 0.5 and $\tilde{V}$ is the posterior probability of the right-most beta mixture given l_{detect} (no threshold). We evaluate $\tilde{U}$ and $\tilde{V}$ every epoch starting at the end of the warm-up phase where the network is trained without correction on the noisy dataset. We end the warm-up phase one epoch after the first learning rate reduction. In summary, OOD noisy labels will be dynamically replaced by a uniform distribution hence promoting their rejection by the network and the clean and corrected ID noisy samples will be assigned a moderately smoothed label, which has been proven to be beneficial for robust DNN training in the presence of label noise [106, 123]. Both $\tilde{U}$ and $\tilde{V}$ are cut of from the computation graph and neither is backpropageted in equation 3.4.

Additional regularization

In order to be competitive with the state-of-the-art, we pair DSOS with two different regularization strategies commonly used to combat label noise. The first regularization we add to the loss promotes high-entropy predictions on ID samples:

$$l_e = -\frac{1}{N}\sum_{i=1}^{N}\tilde{v}_i\sum_{i=1}^{N}h(x_i)\log(h(x_i)). \tag{3.7}$$

We find l_e to be especially important in the warm-up phase as it promotes confident predictions for both the clean samples and the ID samples, which enables better detection. During the label correction phase of DSOS, the regularization is proportionally weighted according to the clean and noisy ID samples detection $\tilde{V}$ so as to not to go against the label softening strategy for OOD samples. We additionally pair DSOS with mixup [227] data augmentation, which has shown to be robust to label noise and that is commonly used in related state-of-the-art noise robust approaches. An ablation study for the different components of DSOS including the effect of the regularizations is given in Section 3.4.3. With $\gamma = 0.4$, the final loss DSOS minimizes is:

$$l = -\frac{1}{N}\sum_{i=1}^{N}y^{d^T}\log(h(x_i)) + \gamma l_e \tag{3.8}$$

3.4 Experiments

3.4.1 Experimental setup

We conduct controlled experiments on corrupted versions of the CIFAR-100 dataset [99] using ImageNet32 [37] images for the OOD noise. The CIFAR-100 dataset is a 32×32 image dataset composed of 50.000 training images and $10,000$ test images, equally distributed over 100 classes. The ImageNet32 dataset is a 32×32 downsized version of the ILSVRC12 [100] dataset $(1.000$ classes and 1.2M images). In order to corrupt CIFAR-100, we consider the OOD noise ratio ρ and the ID noise ratio ψ. We first replace a random fraction ρ of the CIFAR-100 images by randomly selected ImageNet32 [37] images and randomly flip a ψ fraction of the clean samples to a random label assignment. The total noise ratio is $\psi + \rho$. We train for 100 epochs, using a PreActivation ResNet18 [92], SGD with momentum 0.9 and weight decay 5×10^{-4}, starting from a learning rate of 0.03 and reducing it by 10 at epochs 50 and 80, batch size 32 (64 for the warm-up).

For controlled web-crawled datasets, we consider different noise levels $(0\%, 30\%, 50\%, 80\%)$ for the web label noise corruption released for the MiniImageNet ($50k$ training images, $10,000$ test images) and StanfordCars ($8k$ training images, $8k$ test images) datasets [88], adopting the 299×299 image resolution for training and the InceptionResNetV2 network architecture. We train for 200 epochs, using SGD with momentum 0.9 and weight decay 5×10^{-4}, starting from a learning rate of 0.01 and reducing it by 10 at epochs 100 and 160, batch size 32. For real-world web-crawled datasets, we report results training on the mini-Webvision [115] dataset (first 50 classes of WebVision) ($66k$ training images, $2.5k$ test images) at resolution 224×224. We

Table 3.3: DSOS for mitigating ID and OOD noise on CIFAR-100 corrupted with ImageNet32 images. We run each algorithm with the exact same noise corruption. The algorithms we compare with are naive cross-entropy (CE), Mixup (D), Dynamic Bootstrapping (DB), Early Learning Regularization (ELR), EvidentialMix (EDM), Joint Sample Selection and Model Regularization based on Consistency (JoSRC). We report best and last accuracy (best/last).

ρ	ψ	CE	M	DB	ELR	EDM	JoSRC	DSOS		
								ID	OOD	both
0.2	0.2	63.68/55.52	66.71/62.52	65.61/65.61	63.90/63.72	65.11/64.49	67.37/64.17	68.09/67.78	69.37/69.37	70.54/70.54
0.4	0.2	58.94/44.31	59.54/53.16	54.79/54.42	57.16/56.91	55.65/54.49	61.70/61.37	60.12/59.32	62.34/61.03	62.49/62.05
0.6	0.2	46.02/26.03	42.87/40.39	42.50/42.50	31.20/29.55	28.51/10.47	37.95/37.11	46.10/42.93	46.54/40.23	49.98/49.14
0.4	0.4	41.39/18.45	38.37/33.85	35.90/35.90	22.85/21.63	24.15/01.62	41.53/41.44	40.94/35.89	42.53/39.76	43.69/42.88

train for 100 epochs, using an InceptionResNetV2, SGD with momentum 0.9 and weight decay 5×10^{-4}, starting from a learning rate of 0.01 and reducing it by 10 at epochs 50 and 80, batch size 32. We use the mini-WebVision validation set for early stopping and the ILSVRC12 dataset [100] as a test set. For Clothing1M [213] ($1M$ training images, $15k$ test images) we sample 1000 random batches every epoch, resolution 227×227. We train for 100 epochs using a ResNet50 pretrained on ImageNet, SGD with momentum 0.9 and weight decay 1×10^{-3}, starting from a learning rate of 0.002 and reducing it by 10 at epochs 50 and 80, batch size 32. These datasets are common benchmark datasets in the related state-of-the-art and the configurations and networks used follows the state-of-the-art we compare with [88, 111, 119]. A summary of the training details is available in Section 3.4.7.

3.4.2 Experiments on CIFAR-100

We test DSOS in a controlled noise scenario on the CIFAR-100 dataset corrupted with ID symmetric label noise and OOD images from the ImageNet32 dataset in Table 3.3. Contrary to previous works [216], the focus here is on OOD noise. We consider 4 different configurations for CIFAR-100 with $\rho \in [0.2, 0.4, 06]$ and $\psi \in [0.2, 0.4]$. We show the benefits of DSOS when performing ID label bootstrapping or OOD label softening alone as well as the combined benefits of the dual label correction (both in Table 3.3). We compare our approach with two simple baselines: 1) CE, a simple cross-entropy training without any noise correction and 2) mixup (M) [227] a data augmentation strategy robust to label noise. We additionally report results for state-of-the-art noise robust algorithms including Dynamic Bootstrapping (DB) [8] and Early Learning Regularization (ELR) [119]. Finally, we run algorithms focused on OOD and ID noise robustness: EvidentialMix (EDM) [165] and JoSRC [216]. We use the same hyperparameters and network as ours for training the algorithms we compare with except for JoSRC which uses the Adam optimizer by default. A description of the algorithms we compare against is available in Section 2.5. For DSOS, we perform a warm-up training up until after the learning rate reduction. One epoch after the learning rate reduction, we start performing ID and OOD noise detection and apply our label correction strategy with $\alpha = 0.05$. We find that performing warm-up with mixup (M) is better as long as the total noise is superior to 0.8 but use a simple CE warm-up for total noise levels of 0.8. We systematically use the entropy regularization term for the warm-up phase. We report running DSOS with ID or OOD correction alone as well as with combined ID and OOD correction (both). If we notice that the Beta Mixture Model does not capture the ID mode (mode of the first beta distribution outside of the $[0, 1]$ interval) which we observe for total noise levels of 0.8, we fall back to using l_{detect} directly for detecting the ID noisy samples ($l_{detect} < 0.5$ means a samples is ID noisy). We draw the attention of the reader to the improvements

Table 3.4: Ablation study for DSOS. We report best and last accuracy. The baseline is a naive cross-entropy training (CE)

	Best	Last
CE	63.68	55.52
+ mixup	66.71	62.52
+ Entropy regularization	67.27	63.04
+ Batch normalization tuning	67.56	65.69
+ In-distribution bootstrapping	68.09	67.78
+ Out-of-distribution softening	70.54	70.54

DSOS brings when compared to other ID/OOD noise correction approaches even though we use a single network.

3.4.3 Ablation study

We conduct an ablation study to highlight the important elements of DSOS trained on CIFAR-100 with $\rho = 0.2$ and $\psi = 0.2$ (Table 3.4). We find entropy regularization [188] to be necessary to promote confident predictions and specifically study the case where the metrics tracking and the bootstrapped label predictions necessary to applying ID noise correction are computed with trainable batch normalization layers, i.e. the layers get tuned with unmixed samples before evaluation on the validation set. The ablation study highlights how the introduction of the dynamic label softening strategy improves accuracy results over applying ID label correction alone.

3.4.4 Comparison against the state-of-the-art

Table 3.5 reports results for DSOS when compared to state-of-the-art approaches on the web-corrupted versions of Stanford Cars and MiniImageNet [88]. Table 3.6 compares DSOS against state-of-the-art algorithms

Table 3.5: Comparison of DSOS with state-of-the-art algorithms on Mini-ImageNet and Stanford Cars corrupted with web label noise gathered by [88] (red noise). The algorithms we compare with are naive cross-entropy (CE), dropout (D), S-Model (SM), Bootstrapping (B), Mixup (M), MentorNet (MN), MentorMix (MM). We bold best and underline last accuracy for the best performing algorithm.

Dataset	Noise	CE	D	SM	B	M	MN	MM	DSOS
MiniImageNet	0	70.9/68.5	71.8/65.7	71.4/68.4	71.8/68.4	72.8/72.3	71.2/68.9	74.3/73.7	**74.52**/74.10
	30	66.1/56.5	66.6/55.0	65.2/56.3	66.6/56.7	66.8/61.8	66.2/64.0	68.3/67.2	**69.84**/67.86
	50	60.9/51.7	62.1/50.01	61.3/51.3	62.6/52..5	63.2/58.4	61.7/58.0	63.3/61.8	**66.14**/65.18
	80	48.8/39.8	49.5/37.6	49.0/40.6	50.1/40.1	50.7/45.5	49.3/43.4	50.2/48.4	**55.26**/52.24
Stanford Cars	0	90.8/90.8	**92.2**/92.2	90.1/90.1	90.3/90.0	91.9/91.9	90.2/90.1	91.8/91.6	91.38/91.27
	30	80.4/80.2	87.6/87.6	82.2/81.9	83.4/83.0	85.6/85.2	81.1/80.9	87.8/87.7	**88.36**/88.14
	50	70.6/70.3	79.3/79.2	70.1/70.1	73.6/73.5	79.1/78.9	72.0/72.0	80.4/79.8	**82.04**/81.72
	80	43.3/43.0	61.8/61.8	46.4/46.4	47.4/46.7	55.7/55.4	51.0/50.9	58.6/58.6	**62.36**/62.36

on the WebVision 1.0 dataset [115] reduced to the 50 first classes (mini-WebVision, 66K images), a large scale dataset created using web queries. Table 3.7 reports results for Clothing1M. When necessary, we differentiate between methods using a unique network for inference and methods using an ensemble of two networks. In this case, we ensemble two networks trained using DSOS from different random initialization by averaging their prediction at test time and show the direct benefits of using an ensemble in the web label noise scenario. We also notice that DSOS outperforms other algorithms even when no noise is present in Table 3.6. This is most likely due to the our dynamic label softening strategy that has been shown to improve prediction accuracy [133]. We compare with loss or label correction algorithms: Forward correction (**F**) [150], Bootstrapping (**B**) [159], Probabilistic correction (**P**) [217], Joint Optimization (**JO**) [188], S-Model **(SM)** [63]; sample selection algorithms: Co-Teaching (**Co-T**) [68], MentorMix (**MM**) [88], MentorNet (**MN**) [87]; semi-supervised correction algorithm: DivideMix (**DM**) [111], Early Learning Regularization (**ELR** and **ELR+**) [119]; regularization algorithms: Mixup (**M**) [227], Symmetric cross-entropy Loss

Table 3.6: Classification accuracy for DSOS and state-of-the-art methods against methods using a unique network vs an ensemble. We train the network on the mini-Webvision dataset and test on the ImageNet 1k test set (ILSVRC12). We compare with Forward (F), Co-Teaching (Co-T), Mixup (M), MentorMix (MM), Early Learning Regularization (ELR) and DivideMix (DM). All results except our own (DSOS) are from [119]. We bold the best results.

		Unique network						Ensemble of two networks		
		F	Co-T	M	MM	ELR	DSOS	DM	ELR+	DSOS
mini-WebVision	top-1	61.12	63.58	75.44	76.0	76.26	**77.76**	77.32	77.78	**78.76**
	top-5	82.68	85.20	90.12	90.2	91.26	**92.04**	91.64	91.68	**92.32**
ILSVRC12	top-1	57.36	61.48	71.44	72.9	68.71	**74.36**	75.20	70.29	**75.88**
	top-5	82.36	84.70	89.40	**91.10**	87.84	90.80	90.84	89.76	**92.36**

Table 3.7: Comparison of DSOS against state-of-the-art algorithms on Clothing1M. Top-1 best accuracy on the test set. We compare with naive cross-entropy training (CE), Forward (F), Symmetric cross-entropy loss (SL), Joint Optimization (JO), Learning to learn (Me), Probabilistic correction (P), Early Learning Regularization (ELR) and DivideMix (DM). We run ELR and DM using the code provided by the authors. All other results are from the specified works. We bold the best results.

	Unique network								Ensemble of two networks		
	CE	F	SL	JO	ELR	Me	P	DSOS	ELR+	DSOS	DM
Clothing1M	69.10	69.84	71.02	72.16	72.87	73.47	73.49	**73.63**	74.05	74.13	**74.76**

(**SL**) [209]; meta-learning algorithms: Learning to learn (**Me**) [114]; standard cross-entropy training (**CE**), standard cross-entropy plus dropout (**D**). All the algorithms we compare against proposed state-of-the-art results on label noise benchmark datasets at the time of their publication.

3.4.5 Training speed

Table 3.8 reports the wall-clock training time for state-of-the-art methods on the mini-WebVision subset. The first line reports average epoch time,

Table 3.8: Wall-clock training time comparison for state-of-the-art algorithms on the mini-WebVision dataset. All algorithms were run on an RTX 2080 Ti GPU using the PyTorch [149] framework.

	M	ELR	DSOS	ELR+	DM
Epoch	*9.5min*	*10.5min*	*11.25min*	*28min*	*50min*
Full training	*15.75h*	*17.5h*	*18.75h*	*46.75h*	*83h*

warm-up included, and the second line reports the full training duration (100 epochs). Both of these metrics exclude evaluation on a validation set. We compare against state-of-the-art algorithms performing the best on mini-WebVision **DM** [111], **ELR** and **ELR+** [119], **M** [227]. DSOS improves accuracy results on mini-WebVision and trains significantly faster then the closest performing algorithms. Note that the training time for **DM** [111] heavily depends on the training scenario as the algorithm oversamples the unlabeled data every epoch, i.e. the epoch length depends on clean/noisy detection.

3.4.6 Additional explanation of the behavior of the ID/OOD measure

Figure 3.3 illustrates the behavior of our proposed metric l_{detect}. By studying the collision entropy of the interpolated label y_{inter} between the network prediction and the ground truth label, we establish a hierarchy from clean to ID noise to OOD noise. The pivot point $-\log(.5) = 0.693$, computed theoretically when the prediction of the network is absolutely confident but different from the noisy ground-truth, marks the separation between low confidence clean samples and high confidence noisy samples. Although some clean samples will be detected as noisy at the pivot point, because we avoid OOD sample during this transition, we can correct the detected confident

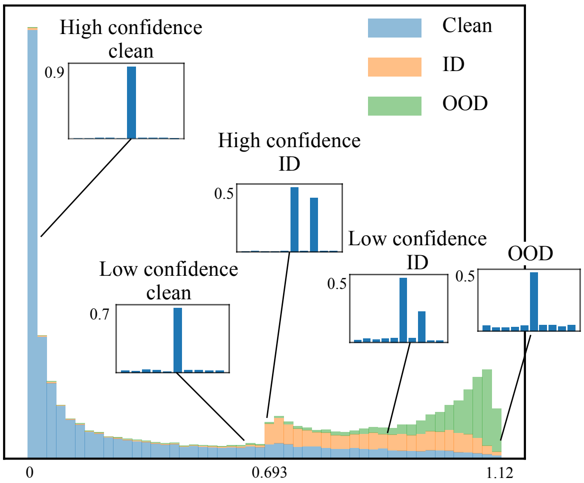

Figure 3.3: Clean/ID/OOD hierarchy established by the collision entropy of the interpolated label l_{detect}

ID samples without concerns of labeling OOD data or corrupting the clean samples since we relabel correct but simply under-confident clean samples: this will not harm the training procedure (their label stays the same). By smoothing OOD samples, we also avoid correcting ID noisy samples with an under-confident corrected prediction.

3.4.7 Hyperparameter table

Table 3.9 details the hyperparameters used in every experiment reported in the state-of-the art comparison. The configuration remains the same across different noise ratios for miniImageNet and Stanford Cars. The parameters common to all experiments are: entropy regularization [188], SGD optimizer, a learning rate decay factor of 10, random horizontal flips, mixup [227] data augmentation. To match the baseline of [88], we add a dropout layer before the fully connected layer in the case of the Stanford Cars experiments. We do not use dropout for other datasets as we manage to match the baselines without it.

Table 3.9: Hyperparameter variations across experiments. We do not change hyperparameters across noise levels for CIFAR-100, mini-ImageNet, and Stanford Cars.

	CIFAR-100	Stanford Cars	miniImageNet	WebVision	Clothing1M
Network	PreActResNet18	InceptionResNetV2	InceptionResNetV2	InceptionResNetV2	ResNet50
ImageNet pretraining	No	No	No	No	Yes
Number of epoch	100	400	200	100	100
Batch size	32	32	32	32	32
Initial learning rate	0.03	0.05	0.01	0.01	0.002
Lr reduction	$[50, 80]$	$[200, 300]$	$[100, 160]$	$[50, 80]$	$[50, 80]$
Weight decay	$5e-4$	$5e-4$	$5e-4$	$5e-4$	$1e-3$
Resize	32	320	320	256	256
RandomResize Range	–	$[0.75, 1.33]$	–	–	–
Crop	32	299	299	227	224
Dropout ratio	0.0	0.3	0.0	0.0	0.0
α	0.05	0.05	0.05	0.05	0.05
Epoch start correction	51	201	101	51	1

3.4.8 Examples of labeled images from the mini-WebVision subset

Figures 3.4 and 3.5 display examples of images labeled from the mini-WebVision subset. The annotations are available together with our code at `github.com/PaulAlbert31/DSOS`. Not all classes present the same quantity of noise for either ID and OOD categories. For example, the Mud Trutle category in Figure 3.5 does not contain ID noise and little OOD noise (only 5 images labeled as OOD over the 3 subsets). Some categories also contain ambiguous OOD noise such as the loggerhead category in Figure 3.5, which contains skeletons of loggerheads annotated as OOD. Whether these examples are relevant or not would depend on the test set. We observe in general that some OOD images in web-crawled datasets are not strictly OOD, meaning that they still share some distant semantics with the target class. In general, it is easy to imagine how the text surrounding some of the noisy images would mention the target category.

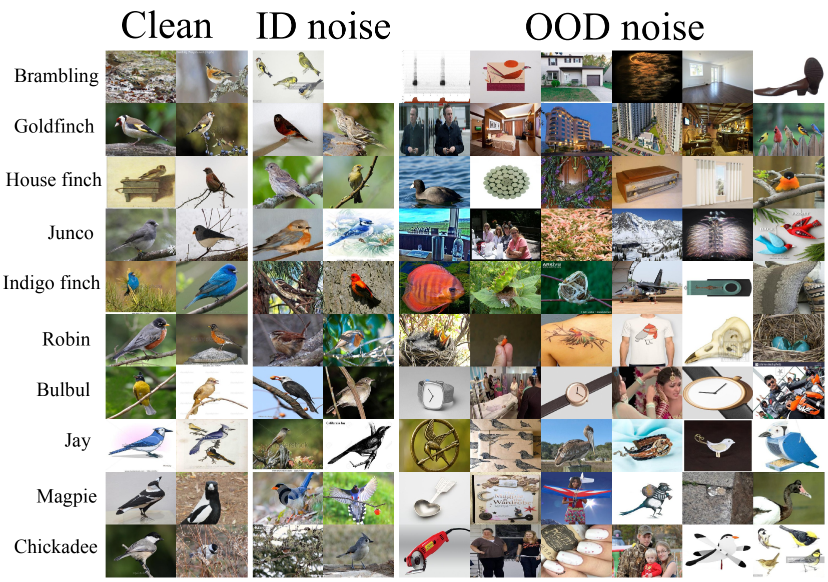

Figure 3.4: First example of samples annotated as clean, in-distribution noise, out-of-distribution noise.

3.4.9 Pseudo-code

Alg. 1 displays pseudo-code for the DSOS algorithm.

3.4.10 Discussion

DSOS improves accuracy results on web crawled datasets such as mini-WebVision (Table 3.6) or web corrupted datasets: miniImageNet (large grained) and Stanford cars (fine grained) in Table 3.5. We explain the lower performance on Clothing1M by the specificity of the gathering process for the dataset, gathered from clothes databases exclusively and not over the whole web and that, according to the authors [213], contains very high levels of in-distribution noise. This goes against our hypothesis of the noise being principally OOD in Section 3.2. Even then, our results are competitive and convergence is reached faster for DSOS, see Table 3.8.

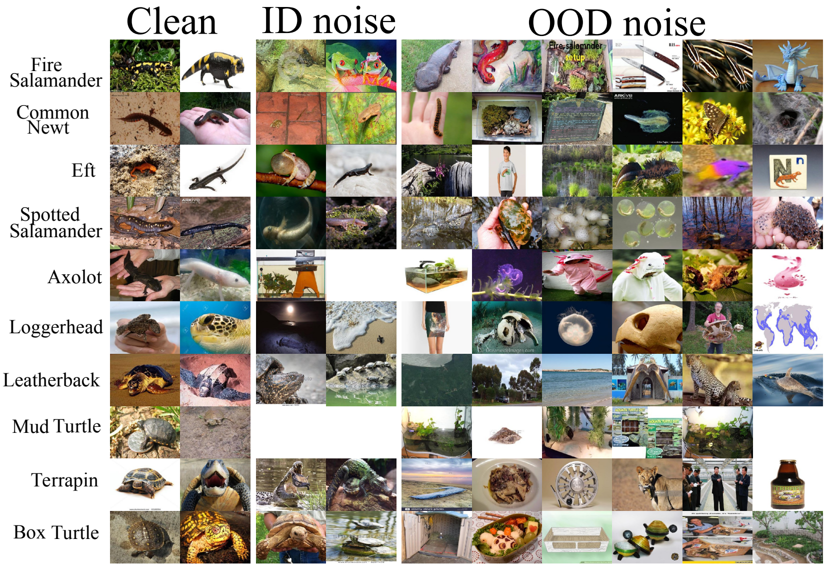

Figure 3.5: More examples of samples annotated as clean, in-distribution noise, out-of-distribution noise.

3.5 Conclusion

The research presented in this chapter aims to address the first research question of this thesis: *"What is the nature of web noise and can detected noisy images be included in the training objective?"*. The nature of noise in web-crawled was shown to be a mixture between in-distribution noise and pre-dominantly out-of-distribution noise. This goes against the common hypothesis of exclusive in-distribution noise that state-of-the-art label noise robust algorithms rely on. To train a neural network on web crawled datasets, we proposed DSOS, a simple algorithm using a novel noise detection metric capable of differentiating between clean, in-distribution noisy and out-of-distribution samples. We propose to detect and treat in-distribution and out-of-distribution noise differently to promote a dynamic rejection of unseen out-of-distribution samples during training, which in turn improves the generalization capabilities of the network. This shows that out-of-distribution images can be used in the training objective to improve the classification accuracy on in-distribution images. Additionally, DSOS is a much simpler approach to label noise than the top state-of-the-art algorithms that we compare against as we use one network and online correction strategy with a single training cycle. By properly identifying and correcting the two distinct label noise distributions, DSOS improves on the most competitive state-of-the-art algorithms. Other strategies could be used to improve network generalization by using out-of-distribution samples such as unsupervised learning, which can learn visual concepts without labels or data augmentation strategies using out-of-distribution samples to efficiently augment in-distribution samples. Because unsupervised learning learns image similarities without the need for labels, it also shows promising perspectives to detect noisy samples in an unsupervised feature space, especially out-of-distribution noise which is visually different from the in-distribution samples. Chapter 4 will study both possibilities.

Algorithm 1 DSOS

Input: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ a web noise dataset. h at convolutional neural network.

Parameters: $\alpha, e_{warmup}, e_{max}$

Output: Trained neural network h_ϕ

```
for $e = 1, \dots e_{warmup}$ do                                   ▷ Warmup
    for $t = 1, \dots numBatches$ do
        Sample the next mini-batch $(x, y)$ from $\mathcal{D}$
        $L = CrossEntro(h(x_{mixed}), y_{mixed})$
        $UpdateNetworkWeights(\mathrm{L})$
    end for
end for
for $e = e_{warmup} + 1, \dots e_{max}$ do                         ▷ Label correction
    $\tilde{U}, \tilde{V}, predictions = EvaluateMetrics(h, \mathcal{D})$    ▷ Evaluated with regards to the original labels
    for $y_i = y_1, \dots y_N$ do
        if $\tilde{u}_i > 0.9$ then    ▷ In-distribution bootstrapping, $\tilde{U} = \{\tilde{u}_i\}_{i=1}^{N}$
            $y_i = p_i$                                    ▷ $predicitions = \{p_i\}_{i=1}^{N}$
        end if
        $y_i = Softmax(y_i v_i / \alpha)$          ▷ Dynamic Softening, $\tilde{V} = \{\tilde{v}_i\}_{i=1}^{N}$
    end for
    for $t = 1, \dots numBatches$ do
        Sample the next mini-batch $(x, y)$ from $\mathcal{D}$          ▷ Train on the corrected labels
        $\tilde{V}_{mini}$ the values in $\tilde{V}$ for the samples in the mini-batch
        $L = CrossEntro(h(x), y)$
        $L = L + 0.4 \times EntroPen(h(x), \tilde{V}_{mini})$        ▷ Weighted entropy penalization
        $UpdateNetworkWeights(\mathrm{L})$
    end for
end for
return $h$                                              ▷ Robustly trained network
```

Chapter 4

Clustering in- and out-of-distribution noise using unsupervised contrastive representations

Chapter 4 continues the research conducted in Chapter 3 on designing algorithms robust to web noise. This chapter proposes to study how unsupervised learning can be used to learn visual features on web noise datasets to detect out-of-distribution images. We show that contrastive learning algorithms trained on web crawled datasets linearly separate in-distribution and out-of-distribution samples on the hyper-sphere projection. Since nothing is known about the noisy nature of images in the web datasets, we cannot estimate the linear separation directly and propose instead to cluster the different type of samples (clean, in-distribution noisy, out-of-distribution) in the contrastive feature space. The proposed algorithm SNCF improves the state-of-the-art classification accuracy on a variety of synthetic benchmarks and the real world WebVision 1.0 web-crawled dataset. Section 4.1 motivates the need to automatically detect in- and out-of-distribution noise in web-crawled datasets, the limitations existing noise-robust algorithms do not address and the research contributions of the chapter. Section 4.2 presents the SNCF algorithm and Section 4.2.3 specifically discuss why out-of-distribution images separates from in-distribution ones in the contrastive feature space. Section 4.3 presents experimental applications of the SNCF algorithm on various noisy image datasets, implementation details, an ablation study, a comparison with the state-of-the-art. Section 4.4 concludes by summarizing the observations made in the chapter and proposes research ideas to improve the SNCF algorithm in the future. The research that emanated from this work was published at the 2022 CVF European Conference on Computer Vision (ECCV).

4.1 Motivations

Designing algorithms capable of training highly accurate CNNs even when trained on imperfect web-crawled data is an important step towards the widespread deployment and take up of computer vision algorithms in practice. CNNs have been shown to completely overfit noisy samples in a dataset without proper regularization [226], which degrades performance. More specifically, chapter 3 observed that the nature of noise in web-crawled datasets can be categorized as both in-distribution and out-of-distribution, the latter being the dominant type. While in-distribution (ID) noisy images can be directly used to train the network after correcting their assigned label, out-of-distribution (OOD) images cannot be assigned to any category. Since a trusted in-distribution dataset is unavailable and the identity of clean and noisy samples is unknown, out-of-distribution detection algorithms [219,

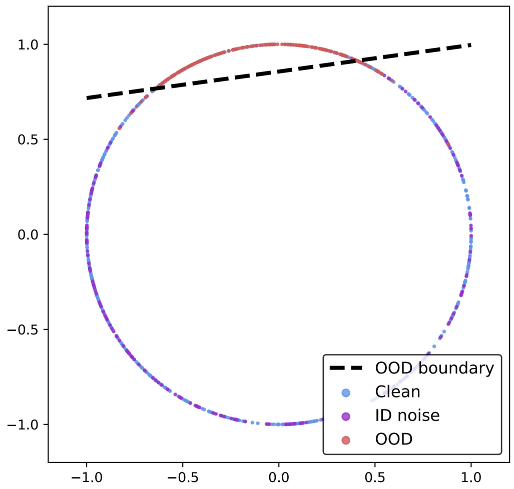

Figure 4.1: Visualization of the linear separation between OOD and ID unsupervised contrastive representations on the 2D hypershpere. CIFAR-10 corrupted with $r_{in} = r_{out} = 0.2$, OOD from ImageNet32. Linear separability in 2D at the dataset level is 92.49% but increases to $98\%+$ for 128D

58, 73], which use a classifier trained on clean data to be able to detect OOD samples post training, cannot be used. This further complicates the noise detection process. Once noisy images have been identified, simply ignoring out-of-distribution images has been shown to be sub-optimal as these samples still contain meaningful information for learning low-level features that can be leveraged to improve the representations learned [78, 216]. This chapter proposes to tackle the in-distribution and out-of-distribution duality of the noise present in web-crawled datasets specifically to improve the final classification accuracy. To detect the noise, we observe that unsupervised contrastive representations for OOD samples become linearly separated from ID ones on the hypersphere (see Figure 4.1) and train a robust network that will use current representations to correct ID noisy samples and use OOD data to improve low-level representations using contrastive learning.

4.2 Algorithm description

This chapter studies image classification in the presence of label noise, where part of the available image dataset $\mathcal{X} = \{x_i\}_{i=1}^N$ and its associated classification labels $\mathcal{Y} = \{y_i\}_{i=1}^N$, with the class distribution $\{c\}_{c=1}^C$, is corrupted by N_{out} out-of-distribution samples and N_{in} in-distribution noisy samples, where $N_c = N - N_{out} - N_{in}$ is the number of in-distribution clean examples. N_{out}, N_{in} as well as the identity of the ID noisy and OOD samples are unknown. Examples of such datasets are web-crawled datasets: WebVision [115], Clothing1M [213], and the Webly Supervised Fine-Grained Recognition datasets [183]. We propose here an algorithm capable of training a convolutional neural network (CNN) h on the corrupted dataset $\mathcal{X}$ without over-fitting to the noise and capable of accurately classifing examples belonging to the class distribution.

4.2.1 Unsupervised feature learning

First, our algorithm learns unsupervised representations from the images themselves, independently of their label. We aim here to relate images to each other in order to capture clusters of similar images. To do so, we train the N-pairs unsupervised contrastive learning algorithm which has been successfully used in metric learning [179] and unsupervised learning on images and text [107]. Given two mini-batches of size B formed from two strongly data augmented views x_i' and x_i'' of $x_i \in \mathcal{X}$, we enforce u_i' and u_i'', their associated contrastive representations through h, to be similar to each other and dissimilar to every other image in the batch. We compute the unsupervised contrastive loss

$$l_{unsup} = -\frac{1}{B}\sum_{i=1}^{B}\log\left(\frac{\exp\left(\mathit{ip}(u''_i, u'_i)/\tau_2\right)}{\sum_{k=1}^{B}\exp\left(\mathit{ip}(u''_k, u'_i)/\tau_2\right)}\right), \tag{4.1}$$

where $\mathit{ip}(u_1, u_2) = \frac{u_1^T.u_2}{\|u_1\|_2\|u_2\|_2}$ is the inner product operation, measuring the similarity between contrastive representations, and τ_2 a temperature hyperparameter, fixed to 0.2 for every experiment. Mixup [227] can be optionally used to further augment x'_i by linearly interpolating it with other augmented samples from the mini-batch with a parameter μ drawn from a beta distribution with parameter 1 to produce $x'_{mix} = \mu x'_i + (1-\mu)x'_j$ with x'_j a random sample from the mini-batch (different for every x_i) and u'_{mix} the associated representation of x'_{mix}. We then use

$$\begin{aligned} l_{mix} = -\frac{1}{B}\sum_{i=1}^{B}\log\Bigg(&\mu\frac{\exp\left(\mathit{ip}(u''_i, u'_{mix})/\tau_2\right)}{\sum_{k=1}^{B}\exp\left(\mathit{ip}(u''_k, u'_{mix})/\tau_2\right)} \\ &+ (1-\mu)\frac{\exp\left(\mathit{ip}(u''_j, u'_{mix})/\tau_2\right)}{\sum_{k=1}^{B}\exp\left(\mathit{ip}(u''_k, u'_{mix})/\tau_2\right)}\Bigg), \end{aligned} \tag{4.2}$$

the N-pairs loss paired with mixup as a data augmentation. This unsupervised contrastive objective has been proposed as part of the iMix [107] algorithm.

4.2.2 Embedding of unsupervised features

We propose not to use the learned unsupervised features directly but instead to perform a non-linear spectral dimensionality reduction on an affinity matrix (embedding). The aim of the embedding is to capture the affinities between samples and their neighbors where ID clean samples will be very similar to other samples from the same class, ID noisy samples will be similar to other ID samples from a different class and OOD samples dissimilar to any ID sample. This motivates computing the embedding at the dataset level to ensure that the similarity of ID noise to other classes is captured. We first compute the sparse similarity matrix S of size $N \times N$ where for each sample in the dataset, we compute the affinity to a fixed neighborhood size of 50 neighbors.

$$S_{ij} = \left(u_i^T u_j / \|u_i\|_2 \|u_j\|_2 \right)^{\gamma}, \tag{4.3}$$

with u_i the unsupervised representation for sample x_i (not augmented) and $\gamma = 3$ a hyper-parameter regulating the importance of distant neighbors. With I_N the identity matrix of size N and D the diagonal normalization matrix where $D_{ii} = \sum_{j=1}^{N} S_{ij}$, we compute the normalized Laplacian

$$L = I_N - D^{-1/2} S D^{-1/2}. \tag{4.4}$$

We finally compute the first k eigenvectors of the normalized Laplacian L by solving

$$(L - \lambda)V = 0 \tag{4.5}$$

and concatenating the first k eigenvectors V of L (by increasing order of the eigenvalues λ, omitting the smallest), providing us with k features per sample to form the embedding E. In practice we use $k = 20$ for every dataset. This embedding process is commonly referred to as spectral embedding [138, 171].

4.2.3 Unsupervised clustering of noise

Using the embedding E, we cluster embedded unsupervised features to identify three kinds of samples: clean ID, noisy ID, and OOD. In the generic case where the three types of noise are expected in the dataset, we apply the clustering at the class level and aim to discover three clusters for each class: a high density cluster of ID clean samples, a low density cluster of OOD samples and ID noisy outliers. In the case where no ID noise is present, we observe the cluster separation at the dataset level and use a two mode Gaussian mixture to retrieve each cluster.

Why does OOD noise cluster? Contrary to previous research where OOD noise is considered an outlier to the distribution [208], we observe in this chapter that unsupervised contrastive learning can be effectively used to cluster noise in the feature space. We expand here on our intuition as to why this works using the alignment and uniformity principles for contrastive learning formalized by Wang et al. [206]. Unsupervised contrastive learning pulls together augmented representations of a same image while pushing apart representations of any other sample in the mini-batch. Since images from a same class will be similar to each other's augmentations, they will cluster together in the feature space to create one (or more) mode for the class (alignment principle). On the other hand, by considering OOD samples as being uniformly sampled from the set of all images, meaning much more varied in appearance than the ID set, we would expect that no compact mode would appear and that these samples would remain uniformly distributed in the feature space yet separated from the ID examples, pulled together into their respective class modes (uniformity principle). Since the features are L^2 normalized during training they exist on the surface of a unit hypersphere and one side of the sphere will contain well represented ID classes, clustered into their respective modes, while OOD noise will remain uniformly distributed be on the other side of the hypersphere and linearly separable from ID samples. Section 4.3.2 proposes experiments to evidence the linear separability of ID and OOD samples in the unsupervised contrastive feature space. The spectral embedding we propose has a key role to play in the clustering of the OOD noise which, although separable from the ID samples, is much more spread-out than the compact class modes of ID images. We remedy this problem by computing the affinities in S not over a fixed distance threshold but using a fixed number of neighbors. Because the affinities in S are then normalized over all neighbors, uniformly distributed OOD samples appear artificially clustered.

It becomes clear that a possible limitation of our approach would be when structure is present in the OOD images i.e. an underling class largely represented in the OOD images, meaning a part of the OOD images are very visually similar to each other and will satisfy the alignment principle of contrastive learning. We do not study this scenario in this chapter since we do not observe it on the web-datasets we train on but recommend further research.

OPTICS [7] is an algorithm which allows us to detect clusters as well as outliers: each feature point is ordered to create a chain where neighboring points are ordered next to each other. Each feature point is then labeled with a reachability cost to neighbors in a neighborhood of size V. The higher the cost, the more likely a sample is to be an outlier. Finally, clusters are identified in the ordering where “valleys” of low reachability cost evidence a cluster, themselves separated with high cost outliers.

Discovering clean and noisy clusters. Because the difficulty of learning similar unsupervised features varies from class to class in an image dataset, we propose to modify the OPTICS algorithm to become more flexible to our problem. We aim here to be able to detect varying valley sizes in the ordered reachability plot where different classes in the image dataset will have more compact classes (fine grained classes) than others (classes with highly diverse examples). In practice, we compute three different reachability orderings for three different neighborhood sizes V $(75, 50, 25$ neighbors), which allows us to account for cluster compactness variations across classes and noise levels. The algorithm chooses the optimal cluster assignment at the class level as being the cluster with the lower amount of outliers given at least two clusters are identified (clean and OOD). This allows us to reduce the amount of hyper-parameters to tune for the clustering to the ξ parameter of OPTICS which controls the decision boundary between clusters and outliers. Higher values for ξ imply a higher tolerance threshold meaning a lower amount of outliers.

No ID noise. In the case where we expect no ID noisy samples in the dataset, we only aim to discover a clean and an OOD cluster without outliers. In this case, the OOD cluster can be retrieved at the dataset level and we choose instead to fit a 2 component Gaussian mixture on the embedded features to retrieve the clusters.

Clean or OOD. Once the clusters are evidenced in the ordering, the final step for the detection is to classify the clusters into clean or OOD. Although the average reachability score within the cluster could at first glance be considered a good indication of the OOD nature of a cluster, by computing the affinity matrix over a fixed neighborhood size, distances are not accurately preserved. We propose instead to compute the density of the cluster in the original unsupervised feature space, where for each sample in the dataset we compute the average distance to all other points in the cluster. We then select the cluster with the lowest density as the OOD cluster.

4.2.4 Spectral Noise clustering from Contrastive Features (SNCF)

Clustering the embedded unsupervised feature space provides three subsets of $\mathcal{X}$: $\mathcal{X}_c, \mathcal{X}_n$ and $\mathcal{X}_o$, respectively the clean, ID noisy and OOD subsets. We aim to use all the available samples to train our CNN and do so by correcting ID noisy samples to their true label and using OOD samples to learn more robust low-level features. We train from scratch on each type of noise separately without using the unsupervised features to initialize the classification network.

Correcting in-distribution noise

Although the unsupervised features allow the detection of incorrectly assigned samples, we find that this is not sufficient to accurately assign ID noise to the right class, especially since they might be close to other ID noisy samples themselves assigned to another incorrect class. We propose instead to correct the ID noise during the supervised training phase, using knowledge learned on clean ID samples during a warm-up pre-training. We then estimate the true labels of the detected ID noise using a consistency regularization approach. For every ID noisy sample in $\mathcal{X}_n$ two weakly augmented versions are produced. The network then predicts on both samples ($p_{i,1}$ and $p_{i,2}$) and returns an average prediction, which, after temperature sharpening τ_1 and normalization, is used as the corrected class assignment: $y_i = \left(\frac{p_{i,1}+p_{i,2}}{2}\right)^{\tau_1}$ with $\tau_1 = 2$ in every experiment. We find temperature sharpening to be necessary to reduce the entropy of the guessed label and to encourage the network to produce more confident predictions.

Out-of-distribution samples

OOD samples cannot be corrected to any label in the distribution but we propose to include them in an additional guided contrastive loss minimization objective to learn low level features. Once the noise detection algorithm has run, we re-embed the unsupervised features of detected OOD noise and use OPTICS to discover clusters of the most similar samples in the OOD data. At training time, we augment each sample in the dataset into one weakly and one strongly augmented view, producing two mini-batches of the same images augmented differently. We then enforce ID samples belonging to the same class (corrected for the ID noise) as well as OOD samples from the same unsupervised cluster to be similar while being dissimilar to every other example in the mini-batch. OOD samples not assigned to any unsupervised cluster are considered similar to their augmented view only. This guided contrastive objective is described in equation 4.7. Note here that the similarities are enforced between the two mini-batches of augmented views alone. We attempted to enforce similarities inside the same augmented batch but noticed no accuracy improvements.

4.2.5 Loss objectives

We consider here that p_i is the current softmax prediction of h on sample $x_i \in \mathcal{X}$. Our algorithm aims to optimize over two objectives during training. The first is the classification objective on the detected clean samples $\mathcal{X}_c$ and the ID samples from $\mathcal{X}_n$ whose label has been corrected. We use the cross entropy loss:

$$l_{ce} = \sum_{i=1}^{N_c+N_n} y_i^T \log(p_i). \tag{4.6}$$

Secondly, we minimize the guided contrastive learning objective, grouping ID samples of the same class and OOD samples from the same OOD cluster together using their respectively weakly and strongly augmented projected representations r_i and r_i', projected from the classification space to the contrastive space

$$l_{cont} = -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{B}\sum_{b=1}^{B} e_{i,b}\log\left(\frac{\exp\left(\boldsymbol{ip}(r_b, r_i')/\tau_2\right)}{\sum_{k=1}^{B}\exp\left(\boldsymbol{ip}(r_k, r_i')/\tau_2\right)}\right), \tag{4.7}$$

with $e_{i,b} = 1$ if sample i is considered similar to sample b, $e_{i,b} = 0$ otherwise. Note that this objective can be paired with mixup as in the unsupervised objective in equation 4.2. The final loss minimized by our algorithm is:

$$l = l_{ce} + \beta l_{cont}, \tag{4.8}$$

where β is an hyper-parameter (typically 1). Figure 4.2 illustrates the algorithm.

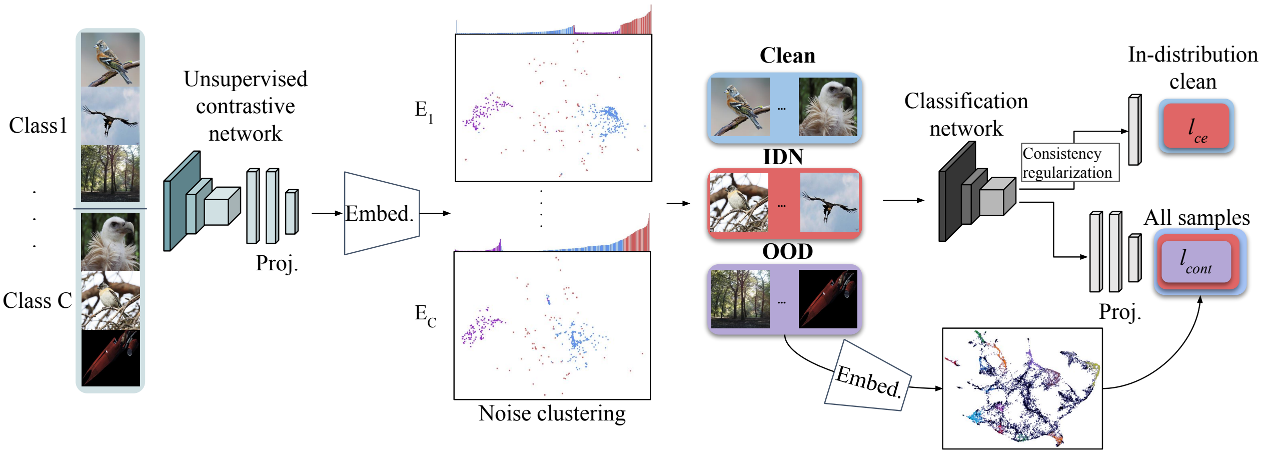

Figure 4.2: Visualization of the algorithm. The unsupervised features are embedded to create E and evaluated at the class level $(E_1, \ldots E_C)$ to cluster clean and OOD samples. The detected OOD samples are re-embedded from their unsupervised features to detect clusters of similar images. We correct the ID noise using consistency regularization and the OOD sample's cluster assignments are used together with the classes of all in-distribution samples in a guided contrastive objective

4.3 Experiments

4.3.1 Implementation details

We form each mini-batch by aggregating an equal number of clean ID, noisy ID and OOD samples. Since the OOD is ignored in the ID objective (eq 4.6) and in order to have the same batch size for the ID and the contrastive forward pass, we form the ID mini-batch by aggregating two weakly augmented views of the clean data with one weakly augmented view for the ID noisy data (ID clean, ID noisy, OOD for the contrastive mini-batch). For the weak data augmentations we use cropping with padding and random horizontal flip and the strong SimCLR augmentations [30]. We warm-up the network on the detected clean data from scratch for 15 epochs in every experiment (except 5 for WebVision) and start both the ID noise correction and the guided contrastive objective after this. For a fair comparison with other approaches, the unsupervised features are not used to initialize the network in the robust classification phase. Since our algorithm minimizes a contrastive loss, we find that adding a non-linear projection head [30, 31] to project features from the classification space to the contrastive space is beneficial in reconciling the training objectives. The final number of projected contrastive features is 128 and the projection head is not used at test time. We use stochastic gradient descent (SGD) with a weight decay of 5×10^{-4} and mixup [227] augmentation with $\alpha = 1$ for all experiments.

Training the unsupervised algorithm. We train the unsupervised algorithm using the same network as the robust classification phase. In cases where the resolution is 227×227 or above, we train and evaluate the unsupervised features at resolution 84×84 as this helps to keep training time and memory consumption reasonable yet still separates the OOD and ID clusters. The algorithm is trained for 2000 epochs, with a batch size of 256, starting with a learning rate of 0.01 and reducing it by a factor of 10 at epochs $1000, 1500$. We use the mixup version on the unsupervised objective (iMix [107]).

Synthetically corrupted datasets. We conduct a first series of experiments on synthetically corrupted versions of CIFAR-100 [99] where we control the ID noise and OOD noise. We use the same configuration as in chapter 3 and note r_{in} and r_{out} the corruption ratios for ID noisy and OOD noise respectively with $r_{in} + r_{out}$ the total noise level. Our focus here is on the OOD noise rate more than ID noise, which is less present in web-crawled datasets. We introduce OOD noise by replacing original images with images from another dataset, either ImageNet32 [37] or Places365 [230]. For the ID noise, we randomly flip the labels of a portion of the dataset to a random label (uniform noise). The dataset size remains 50K images after noise injection. We train on CIFAR-100 using a PreActResNet18 [92] trained with a batch size of 256 for 100 epochs with a learning rate of 0.1, reducing it by a factor of 10 at epochs 50 and 80.

Web noise corruption. We conduct experiments on miniImageNet [202] corrupted by web noise from Jiang et al. [88] (Controlled Noisy Web Labels, CNWL) where the severity of the web noise corruption is controlled. This dataset is an example where ID noise is very limited and where we find that using the 2 components Gaussian Mixture Model is sufficient to detect the noise at the dataset level (see Section 4.2.3). We train on this dataset at two different resolutions, first 299×299, which is the original configuration proposed by Jiang et al. [88] and second the 32×32 resolution adopted in recent works [39, 215, 166]. For the 299×299 configuration, we train an InceptionResNetV2 [185] with a batch size of 64 for 200 epochs with a learning rate of 0.01, reducing it by a factor of 10 at epochs $100, 160$. For the 32×32 configuration, we use the same configuration as CIFAR-100.

Real-world dataset. We evaluate our model on the (mini)WebVision [115] dataset reduced to the first 50 classes (65k images). We train an InceptionResNetV2 [185] at a 227×227 resolution with a batch size of 64 for 100 epochs with a learning rate of 0.01, reducing it by 10 at epochs $50, 80$.

Baselines. We introduce here the state-of-the-art approaches we compared with as well as the abbreviations used in the tables. Cross-entropy (CE), dropout (D), and mixup (M) are simple baselines obtained by training with no noise correction and dropout [10] or mixup [227] as regularization. MentorNet [87] (MN) and MentorMix [88] (MM) use teacher networks to weight noisy samples. FaMUS [215] (FaMUS) uses meta learning to learn to correct noisy samples. Bootstrapping [159] (B) corrects noisy samples using a fixed interpolation with pseudo-labels; Dynamic Bootstrapping [8] (DB) expands the idea by correcting only high loss noisy samples retrieved using a beta mixture. The S-model [63] (SM) corrects noisy samples using a noise adaptation layer optimized using an Expectation Maximization (EM) algorithm. DivideMix [111] (DM) uses a Gaussian mixture to detect high loss samples and correct them using a semi-supervised consistency regularization algorithm; the idea is expanded upon in PropMix [39] (PM) where unsupervised initialization is used and only the simplest of the noisy samples are corrected while the hardest are discarded. ScanMix [166] (SM) also improves on DM by correcting the label using a semi-supervised contrastive algorithm together with a semantic clustering in an unsupervised feature space, optimized using an EM algorithm. EvidentialMix [165] (EDM) refines the noisy sample detection of DM to account for OOD samples and uses the evidential loss [169] to evidence separate OOD and ID noisy modes. JoSRC [216] (JoSRC) proposes to use the Jensen-Shannon divergence between a consistency regularization guessed label and the original label to detect noisy samples and further select samples with low agreement between views as OOD. Robust Representation Learning [113] (RRL) trains a weakly supervised prototype objective to promote clean samples to be close to their class prototypes. Finally, Dynamic Softening for Out-of-distribution Samples (DSOS) is presented in chapter 3 and computes the collision entropy of the interpolation between the original label and network prediction to separate ID noisy and OOD samples.

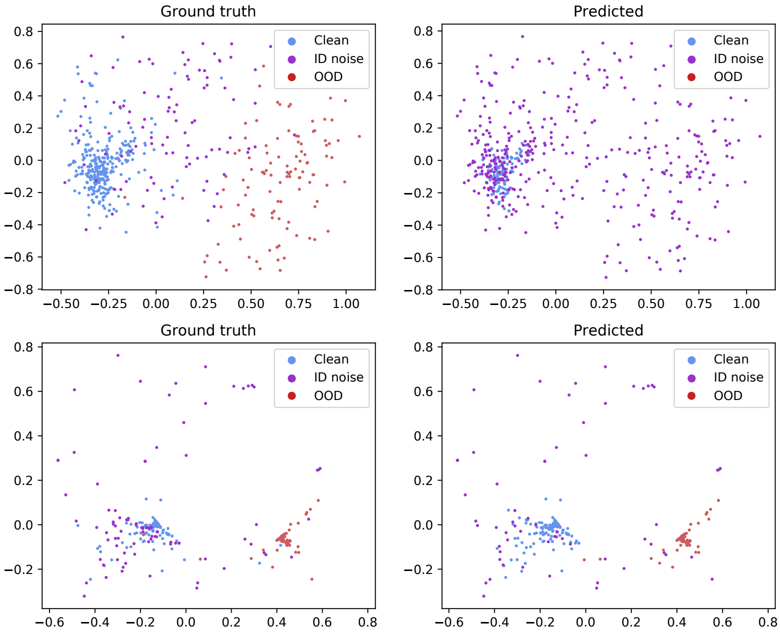

Figure 4.3: Feature embedding for class 1 of CIFAR-100 corrupted with $r_i = r_o = 0.2$ (ImageNet32 OOD). The top row presents a 2D visualization obtained using Isomap [191] of the raw contrastive features and the second row presents an Isomap visualization of the embedding E. Embedding the features allows us to evidence the OOD cluster

4.3.2 Clustering the unsupervised features

This section presents the experiments on the linear separability of ID and OOD data in an unsupervised feature space, the importance of embedding the unsupervised features when clustering the noise and the accuracy of our noise retrieval algorithm. First, to validate our hypothesis over the separability of ID and OOD samples in the unsupervised feature space, we propose to train a linear classifier on the unsupervised features using oracle clean and noisy labels to evaluate a linear separation score for the two distributions. We observe that the classifier can linearly separate the two distribution with error rates below 3% for synthetically corrupted CIFAR-100 with $r_{in} = r_{out} = 0.2$ and below 1% for miniImageNet corrupted with web noise (CNWL). The linear separability is less accurate when using Places365

Table 4.1: Mitigating ID noise and OOD noise on CIFAR-100 corrupted with ImageNet32 or Places365 images. We run all the algorithms using publicly available implementations by authors. We compare with naive cross-entropy training (CE), Mixup (M), Dynamic Bootstrapping (DB), Joint Sample Selection and Model Regularization based on Consistency (JoSRC), Early Learning Regularization (ELR), EvidentialMix (EDM), Chapter 3 (DSOS), Robust Representation Learning (RRL). We report best and last accuracy. We bold (underline) the highest best (final) accuracy

Corruption	r_{out}	r_{in}	CE	M	DB	JoSRC	ELR	EDM	DSOS	RRL	Ours
INet32	0.2	0.2	63.68/55.52	66.71/62.52	65.61/65.61	67.37/64.17	68.71/68.51	71.03/70.42	70.54/70.54	72.64/72.33	**72.95**/72.70
	0.4	0.2	58.94/44.31	59.54/53.16	54.79/54.42	61.70/61.37	63.21/63.07	61.89/61.83	62.49/62.05	66.04/65.44	**67.62**/67.14
	0.6	0.2	46.02/26.03	42.87/40.39	42.50/42.50	37.95/37.11	44.79/44.60	21.88/14.59	49.98/49.14	26.76/24.51	**53.26**/51.26
	0.4	0.4	41.39/18.45	38.37/33.85	35.90/35.90	41.53/41.44	34.82/34.21	24.15/01.62	43.69/42.88	31.29/30.64	**54.04**/52.66
Places365	0.2	0.2	59.88/53.61	66.31/59.69	65.86/65.83	67.06/66.73	68.58/68.45	70.46/70.25	69.72/69.12	**72.62**/72.49	71.25/71.14
	0.4	0.2	53.46/42.46	59.75/48.55	55.81/55.61	60.83/60.64	62.66/62.34	61.80/61.55	59.47/59.47	**65.82**/65.79	64.03/63.48
	0.6	0.2	39.55/21.42	39.17/33.69	40.75/40.61	39.83/39.63	37.10/36.51	23.67/14.66	35.48/35.41	49.27/49.27	**49.83**/49.83
	0.4	0.4	32.06/13.85	34.36/27.63	35.05/34.86	33.23/32.58	34.71/33.86	20.33/11.88	29.54/29.48	26.67/24.34	**50.95**/47.61

as the OOD corruption dataset; we argue that this is because of lower image variability in the dataset, justified by the lower number of classes and the fine-grained nature of the classification task. Second, Figure 4.3 provides a visualization of the importance of embedding the unsupervised features to perform the noise clustering for a class of CIFAR-100 where we compare applying the clustering algorithm on the raw unsupervised contrastive features against the spectral embedding E. The left column is the ground-truth and the right represent the detection made by the clustering algorithm. We use Isomap [191] to reduce the dimentionality to 2 to be able to visualize the features. The spectral embedding E is essential to evidence the OOD cluster, not originally present in the raw features.

4.3.3 Synthetic noise corruption

We study the capacity of our algorithm to mitigate ID noise and OOD noise on synthetically corrupted version of the CIFAR-100 dataset. Table 4.1 reports

Table 4.2: Ablation study on CIFAR-100 corrupted with ImageNet32 with $r_{out} = 0.4$ and $r_{in} = 0.2$. corr = correction and rm = remove

		Embed	Contrastive	Best	Last
No noise corr	CE	✗	✗	58.94	44.31
	+ mixup	✗	✗	59.54	53.16
	+ guided contrastive	✗	✓	62.83	56.29
Noise corr	ID corr only	✗	✗	57.02	55.43
	rm OOD only	✗	✗	60.73	53.88
	ID corr and rm OOD	✗	✗	54.81	54.20
	ID corr only	✓	✗	61.40	58.90
	rm OOD only	✓	✗	60.87	54.08
	ID corr and rm OOD	✓	✗	61.83	61.45
	ID corr only	✓	✓	63.91	62.94
	ID corr and rm OOD	✓	✓	64.51	64.04
	OOD corr only	✓	✓	63.41	58.39
	ID + OOD corr	✓	✓	65.22	64.42
Other	+ equal sampling	✓	✓	67.62	67.14
	- mixup	✓	✓	61.66	59.40

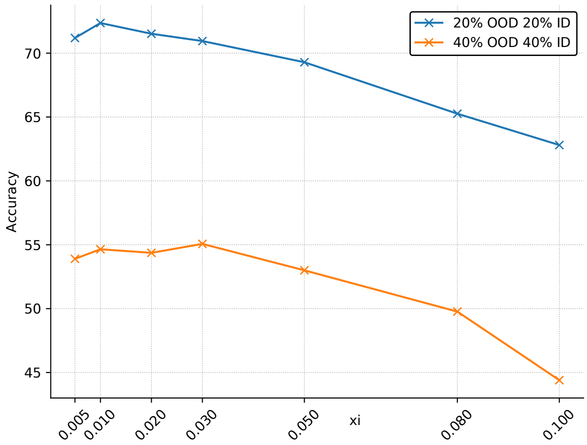

Figure 4.4: Hyper-parameter tuning for OPTICS. We report accuracy results obtained for ID/OOD clustering setting different ξ values in OPTICS

results when using ImageNet32 or Places365 as a OOD corruption. We notice that the OOD corruption using the Places365 dataset is more harmful than corrupting with ImageNet32, especially for high noise levels.

4.3.4 Ablation study

Table 4.2 illustrates the importance of each element of the proposed method on CIFAR-100 corrupted with OOD noise from ImageNet32. We study multiple

Table 4.3: Web-corrupted miniImageNet from the CNWL [88] (32×32). We run our algorithm; other results are from [39]. We denote with $\star$ algorithms using an ensemble of networks to predict and with † algorithms using unsupervised initialization. We compare with naive cross-entropy training (CE), Mixup (M), DivideMix (DM), MentorMix (MM), Fast Meta Update Strategy (FaMUS), ScanMix (SM), PropMix (PM). We report best accuracy and bold the best results

Noise level	CE	M	$\star$DM	MM	FaMUS	$\star$†SM	$\star$†PM	Ours
20	47.36	49.10	50.96	51.02	51.42	59.06	61.24	**61.56**
40	42.70	46.40	46.72	47.14	48.03	54.54	56.22	**59.94**
60	37.30	40.58	43.14	43.80	45.10	52.36	52.84	**54.92**
80	29.76	33.58	34.50	33.46	35.50	40.00	43.42	**45.62**

cases including retrieving OOD and ID clusters on the un-embedded raw unsupervised contrastive features (Noise corr without embedding); correcting only the OOD or ID examples while considering the rest clean (ID/OOD corr only); joint effect of the ID and OOD correction (ID + OOD corr); studying the effect of removing the OOD samples from the training set instead of using them in the guided contrastive objective in equation 4.7 (ID corr and rm OOD); runing the algorithm without mixup (- mixup). We point out how important mixup is (especially in the classification loss) to avoid overfitting to the noise. Figure 4.4 reports the quality of our robust classification algorithm on ID and OOD clustering for different values of ξ in OPTICS where values inferior to 0.03 lead to the best results. We choose $\xi = 0.01$ for all datasets.

4.3.5 Results on web-noise

We consider here the controlled noisy web labels (CNWL) dataset, where miniImageNet is corrupted with OOD images from web queries. Table 4.3 reports results when training at resolution 32×32 and Table 4.4 at resolution 299×299. Finally, in Table 4.5 we train on the first 50 classes of the

Table 4.4: Web noise on CNWL [88] trained at a high resolution (299 × 299). We run our algorithm, other results are from Chapter 3. We compare with naive cross-entropy training (CE), Dropout (D), S-Model (SM), Boostrapping (B), Mixup (M), MentorNet (MN), MentorMix (MM), Chapter 3 (DSOS). We report best accuracy

Noise level	CE	D	SM	B	M	MN	MM	DSOS	Ours
0	70.9/68.5	71.8/65.7	71.4/68.4	71.8/68.4	72.8/72.3	71.2/68.9	74.3/73.7	74.52/74.10	**74.80**/74.60
30	66.1/56.5	66.6/55.0	65.2/56.3	66.6/56.7	66.8/61.8	66.2/64.0	68.3/67.2	69.84/67.86	**69.96**/69.64
50	60.9/51.7	62.1/50.01	61.3/51.3	62.6/52..5	63.2/58.4	61.7/58.0	63.3/61.8	66.14/65.18	**66.48**/66.38
80	48.8/39.8	49.5/37.6	49.0/40.6	50.1/40.1	50.7/45.5	49.3/43.4	50.2/48.4	55.26/52.24	**55.54**/54.96

WebVision dataset (mini-WebVision) a real world web-crawled dataset and report top-1 and top-5 accuracy results on the validation set on WebVision and on the test set on the ImageNet1k (ILSVRC12) dataset. Since our algorithm uses only one network and to compare against ensemble methods, we report an additional result where we ensemble two networks trained from different random initializations by averaging their prediction at test time. We also report results when training for 150 epochs to compare fairly against FaMUS. Our algorithm slightly outperforms the state-of-the-art for top-1 accuracy but more convincingly so for top-5 accuracy on WebVision. Because of the guided contrastive loss, the network learns more generalizable features which reduce the risk of catastrophic classification errors (when the predicted class is completely semantically different from the correct predictions). mini-Webvision in particular proposes fine grained classification on species of birds, amphibians and marine animals which reward generalizable features for top-5 evaluation.

Table 4.5: Classification accuracy for the proposed and other state-of-the-art methods. We denote with $\star$ algorithms using an ensemble of networks to predict and with † algorithms using unsupervised initialization. We train the network on the mini-Webvision dataset and test on the ImageNet 1k test set (ILSVRC12). We compare with Mixup (M), MentorMix (MM), DivideMix (DM), Early Learning Regularization (ELR), Robust Representation Learning (RRL), Chapter 3 (DSOS), PropMix (PM), ScanMix (SM), Fast Meta Update Strategy (FaMUS). We bold the best results

		100 epochs										150 epochs	
		M	MM	$\star$DM	$\star$ELR+	RRL	$\star$DSOS	$\star$†PM	$\star$†SM	Ours	$\star$Ours	FaMUS	$\star$Ours
mini-WebVision	top-1	75.44	76.0	77.32	77.78	77.80	78.76	78.84	**80.04**	78.16	79.84	79.40	**80.24**
	top-5	90.12	90.2	91.64	91.68	91.30	92.32	90.56	93.04	92.60	**93.64**	92.80	**93.44**
ILSVRC12	top-1	71.44	72.9	75.20	70.29	74.40	75.88	−−	75.76	74.20	**76.64**	77.00	**77.12**
	top-5	89.40	91.10	90.84	89.76	90.90	92.36	−−	92.60	93.32	**94.20**	92.76	**94.32**

Table 4.6: Linear separation between ID and OOD noise in contrastive and non-contrastive algorithms. CIFAR-100 with $r_{out} = 0.4$ and $r_{in} = 0.2$ ImageNet32 corruption.

	N-pairs	iMix	lunif+lalign	SimCLR	Mocov3	BYOL	SimSiam	Barlow Twins	VicReg	DeepClusterV2	Swav
linear sep	94.77	95.61	94.14	95.47	98.15	98.69	94.50	98.59	98.57	87.40	85.74
kNN acc	55.29	53.98	55.09	55.36	53.13	60.54	56.52	59.30	58.61	53.50	52.21

4.3.6 Alternative contrastive algorithm

We study here how well the linear separation we observe with the N-pairs algorithm translates to recent state-of-the-art contrastive and non-contrastive algorithms. For the contrastive algorithms we study N-pairs, iMix [107], SimCLR [31], the combination of alignment and uniformity losses from Wang et al. [206] and Mocov3 [33]. For the a-priori non-contrastive approaches, we study BYOL [65], SimSiam [32] and Barlow Twins [222]. We also add some clustering-based algorithms: DeepClusterv2 [25] and Swav [26]. As stated in section 4.2.3, we expect that non contrastive algorithms will exhibit a degraded linear separation. We use the same code base for all algorithms [40] and all approaches use a non-linear projection head and L2 normalization of features. Table 4.6 reports the linear separation results and kNN accuracy (k=200). We observe that the linear separation between OOD and ID samples is independent from the kNN accuracy as Mocov3 and BYOL have similar linear separation scores but BYOL is much more accurate at kNN classification. Both a-priori contrastive and non-contrastive approaches demonstrate good separability and clustering approaches propose a much worse separation. A priori non-contrastive approaches (BYOL, SimSiam, Barlow Twins) have been recently shown to be hidden, dimension contrastive algorithms by Garrido et al. [60] as opposed to sample contrastive approaches (SimCLR, N-pairs). We believe this might explained why the linear separation is observed for the a-priori non-contrastive algorithms (see Section 4.2.3). It is also interesting to note that unsupervised algorithms generalize differently to real world data. From a kNN accuracy point of view, some approaches maintain the accuracy improvement on noisy datasets (Barlow Twins, BYOL) while others are affected by the noise present in the dataset (SimSiam, iMix, Mocov3). Since large web datasets present a great opportunity for unsupervised feature learning, we believe that robustness to label noise should be of interest in future unsupervised representation learning research.

4.3.7 Discussion on SNCF compared to DSOS

Both DSOS in Chapter 3 and SNCF in this chapter aim to address the same problem of learning an accurate classifier on noisy data. Comparisons conducted in Tables 4.4 and 4.5 show that SNCF is capable of using unsupervised features and contrastive learning to achieve a higher classification accuracy than DSOS in web-noise benchmark datasets. The improvements over DSOS are explained by the improved detection of the OOD images where the separation between ID and OOD in SNCF is almost perfect in simpler datasets (see Table 4.6) and also because training the guided contrastive objective together with the classification one allows us to learn better features that boost the classification accuracy. In Table 4.2, adding the guided contrastive objective improves the classification accuracy by approximately 2 points.

4.4 Conclusion

This chapter aims to address research question two of this thesis: *"Can unsupervised learning be used to detect noise in web-crawled datasets?"*. We found that we can use the alignment and uniformity principles of unsupervised contrastive learning to detect ID and OOD label noise clusters in an embedded feature space. We show that the unsupervised contrastive features for OOD and ID samples are, to a large extent, linearly separated on the unit hypersphere and compute a fixed neighborhood spectral embedding to reduce differences in cluster densities. We adapt the OPTICS algorithm, ordering samples in a neighbor chain and computing the reachability cost to neighbors. Clusters are evidenced by valleys in the reachability plot and a voting system automatically selects the best cluster assignment at the class level given multiple neighborhood sizes. Once the noise has been identified, we train a robust classifier that corrects the labels of known ID noisy samples using a consistency regularization estimation and uses ID and OOD samples together in an auxiliary guided contrastive objective. This completes the answer to the first research question given in Chapter 3 on the usability of OOD images to improve ID classification. We report state-of-the-art results on a variety of noisy datasets including synthetically corrupted versions of

CIFAR-100, controlled web noise in miniImageNet, and mini-WebVision as a real-world web-crawled dataset.

Chapters 4 studied how unsupervised learning could be used to detect noisy samples in web crawled datasets which require little human supervision. An alternative approach to reduce human supervision is semi-supervised learning where only part of the images are labeled by humans. Because most images are unlabeled in semi-supervised datasets, unsupervised learning has a strong potential to improve classification accuracies in this scenario. Chapter 5 will study how unsupervised learning can be used as a mean to propagate labels from the labeled to unlabeled set of a semi-supervised dataset.

Chapter 5

Unsupervised learning to bootstrap additional labels for semi-supervised learning

This chapter presents an approach to semi-supervised learning that is designed to improve the performance of semi-supervised algorithms when a small amounts of labeled data is available. The approach makes use of unsupervised features to learn useful features for the unlabeled data. These features are in turn used for propagating the few known labels and a large trusted subset is extracted to be used for semi-supervised learning. The proposed knowledge bootstrapping pipeline does not require additional supervision outside of the few initially labeled data. We substantially improve semi-supervised errors in scenarios of less than 25 samples per class on CIFAR and miniImageNet. Section 5.1 motivates the semi-supervised problem we tackle in this chapter and the proposed solution. Section 5.2 formalizes ReLaB, the proposed algorithm. Section 5.3 reports experiment results for different unsupervised algorithms, shows the benefits of ReLaB over other noise-robust algorithms when dealing with the noisy dataset resulting from the label propagation, and tests our approach for different amounts of labeled samples on CIFAR10, CIFAR100, and miniImageNet. Section 5.4 concludes the chapter. The research that emanated from this work was published in the 2021 International Joint Conference on Neural Networks (IJCNN).

5.1 Motivations

Despite recent efforts in the semi-supervised learning literature aiming at reducing human supervision further, extreme label scarcity is still challenging [19, 178]. In the absence of labels, the unsupervised paradigm for unsupervised visual representation learning has recently gained traction [11, 44, 57, 61, 30]. Unsupervised learning constructs a supervisory signal using a pretext task where pretext labels are generated from the data. By solving pretext tasks such as colorization of greyscale images [229], prediction of image rotations [61], or contrasting different views of the same image [30], high quality features can be learned without human annotations. The success of unsupervised learning has motivated its adoption for semi-supervised learning, which improved performance in cases of very low label availability [207, 19]. Berthelot et al. [19] and Wang et al. [207] use unsupervised regularization which stabilizes network training, while Rebuffi et al. [158] make use of unsupervised pre-training [61] as an initialization strategy for semi-supervised training.

This chapter explores the idea of automatically annotating images using label propagation. In particular, we use representations learned by unsupervised tasks together with a low amount of labels to apply label propagation and spread the available labels to the entirety of the samples. This results in a fully labeled dataset which contains numerous incorrect (noisy) annotations. We then select a trusted, clean subset from this noisy dataset that reliably extends the initially labeled data. The extended labeled dataset is then used to enhance the performance of any semi-supervised image classification algorithm when very few labeled samples are available. We name this label bootstrapping strategy ReLaB. When ReLaB is used to bootstrap labels for ReMixMatch [19] on CIFAR-10 with 10, 40, 100 labeled samples, we reduce the accuracy error by more than 36, 22, 15 absolute percentage points respectively. ReLaB's unsupervised knowledge-bootstrapping pipeline makes use of unsupervised, image retrieval and label noise solutions to provide an approach for scenarios of extremely scare annotations in semi-supervised learning. This could include visual domains where annotations are either time-consuming and expensive to gather or when expert annotators are required.

5.2 Reliable label bootstrapping for semi-supervised learning

We formulate a semi-supervised classification task for C classes as learning a model h given a training set $\mathcal{D}$ of N samples. The dataset consists of the labeled set $\mathcal{D}_l = \{(x_i, y_i)\}_{i=1}^{N_l}$ with corresponding one-hot encoded labels $y_i \in \{0,1\}^C$ and the unlabeled set $\mathcal{D}_u = \{x_i\}_{i=1}^{N_u}$, $N = N_l + N_u$ the total number of samples. We consider a CNN for $h : \mathcal{D} \rightarrow [0,1]^C$. The network is comprised of a feature extractor $h_f : \mathcal{D} \rightarrow \mathcal{F}$, mapping the input space to the feature space $\mathcal{F}$, and a classifier $h_c : \mathcal{F} \rightarrow [0,1]^C$.

We address the case where $\mathcal{D}_l$ contains a low amount of samples. Con-

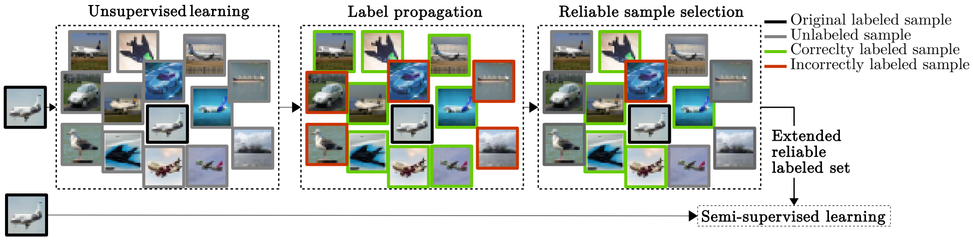

Figure 5.1: Reliable Label Bootstrapping (ReLaB) overview on CIFAR-10 (best viewed in color). Unlike traditional SSL (bottom) that directly uses the labeled examples provided (*airplane*), ReLaB (top) bootstraps additional labels before applying SSL.

trary to usual semi-supervised algorithms, before starting to train the neural network for classification, we propose to extend $\mathcal{D}_l$ to a larger dataset $\mathcal{D}_r$ of size $N_r > N_l$ by automatically labeling samples from $\mathcal{D}_u$. We do so, by propagating labels from $\mathcal{D}_l$ to $\mathcal{D}_u$ using the unsupervised features learned on $\mathcal{D}$. In order to avoid overfitting to incorrect class assignments computed in the label propagation phase, we build $\mathcal{D}_r$ by selecting clean (reliable) samples from the propagated labels. This is done using label noise methodologies. Training on $\mathcal{D}_r$ instead of $\mathcal{D}_l$ greatly improves the performance of semi-supervised algorithms when very few labels are available. Figure 5.1 presents and overview of our proposed approach.

5.2.1 Label propagation on unsupervised features

Knowledge transfer from the labeled set $\mathcal{D}_l$ to the unlabeled set $\mathcal{D}_u$ is implicitly done by semi-supervised learning approaches as the network predictions for $\mathcal{D}_u$ can be seen as estimated labels. With few labeled samples however, it is difficult to learn useful initial representations from $\mathcal{D}_l$ and performance is substantially degraded [19] (see Subsection 5.3.5).

Conversely, we propose to learn a set of descriptors in an unsupervised manner and subsequently propagate the labels on the data manifold, in order to retrieve additional labels for the unlabeled data.

We adopt the established graph diffusion algorithm [81, 46, 186, 80, 197] for label propagation. We formulate the label propagation problem in a similar fashion than [81] except that we study the estimation of $\hat{y}$ as a label propagation task using unsupervised visual representations learned from all samples in $\mathcal{D}$. In particular, we learn a feature extractor h_{φ_f} using unsupervised learning to obtain class-discriminative image representations [97] and subsequently propagate labels from the N_l labeled images to estimate labels $\hat{y}$ for the N_u unlabeled samples. We do so by solving a label propagation problem based on graph diffusion [81]. First, the set of descriptors $\{v_i\}_{i=1}^{N}$ are used to define the affinity matrix:

$$S = D^{-1/2} A D^{-1/2}, \tag{5.1}$$

where $D = \operatorname{diag}\left(A\mathbb{1}_N\right)$ is the degree matrix of the graph and the adjacency matrix A is computed as $A_{ij} = \left({v_i^T v_j}/{\|v_i\|\|v_j\|}\right)^{\gamma}$ if $i \neq j$ and 0 otherwise. γ weighs the affinity term to control the sensitivity to far neighbors and is set to 3 as in [81]. The diffusion process estimates the $N \times C$ matrix as:

$$F = \left(I - \alpha S\right)^{-1} Y, \tag{5.2}$$

where α denotes the probability of jumping to adjacent vertices in the graph and Y is the $N \times C$ label matrix defined such that $Y_{ic} = 1$ if sample $x_i \in \mathcal{D}_l$ and $y_i = c$ (i.e. belongs to the c class), where i (c) indexes the rows (columns)

Table 5.1: Class and noise imbalance after applying label propagation

	CIFAR-10		CIFAR-100	
$\frac{N_l}{C}$	#sample	noise ratio	#sample	noise ratio
4	4249 ± 1726	24.14 ± 10.42	472 ± 161	50.52 ± 16.79
10	4888 ± 1367	24.28 ± 7.43	477 ± 180	39.92 ± 15.31
25	4990 ± 1036	9.50 ± 6.90	444 ± 233	33.39 ± 12.55

in Y. Finally, the estimated one-hot label $\hat{y}_i$ is:

$$\hat{y}_{ic} = \begin{cases} 1, & \text{if } c = \arg\max_c \ F_{ic} \\ 0, & \text{otherwise} \end{cases},$$

for each unlabeled sample $x_i \in \mathcal{D}_u$. The estimated labels allow the creation of the extended dataset with estimated noisy labels $\hat{\mathcal{D}} = \{(x_i, \hat{y}_i)\}_{i=1}^{N}$, where $\hat{y}_i = y_i, \forall\, x_i \in \mathcal{D}_l$.

5.2.2 Reliable sample selection: dealing with noisy labels

Propagating existing labels using unsupervised representations as described in Section 5.2.1, results in estimated labels $\hat{y}_i$ that might be incorrect, i.e. label noise. Using noisy labels as a supervised objective on $\hat{\mathcal{D}}$ leads to performance degradation due to label noise memorization [226, 81]. Since the label noise in $\hat{\mathcal{D}}$ comes from features extracted from the data, noisy samples tend to be visually similar to the seed samples which poses a challenging scenario as noise-robust, state-of-the-art training strategies [8, 119, 227] experience important limitations (see Table 5.4).

Moreover, we find that this label noise is unbalanced in terms of number of samples and different levels of noise in each class. We report in Table 5.1 the median and standard deviation for the number of sample per class (#samples) and noise ratio over the classes of CIFAR-10 and CIFAR-100 for different amounts of labeled samples in N_l. Using the small loss trick to select a subset of clean samples is commonly used in the label noise literature [43, 96, 144, 145], but the issues specific to label noise resulting from label propagation are not addressed in the label noise literature and pose additional challenges, see Section 5.3.3.

In particular, we identify clean samples using the cross-entropy loss:

$$\ell_i = -\hat{y}_i^T \log h(x_i), \tag{5.3}$$

with softmax-normalized logits $h(x_i)$ and training with a high learning rate (small loss) which helps prevent label noise memorization [8] on the extended dataset $\hat{\mathcal{D}}$. The reliable set $\mathcal{D}_r = \{(x_i, \hat{y}_i)\}_{i=1}^{N_r}$, with $N_r > N_l$, is then created by selecting for each class c the N_l^c originally labeled samples for that class c in $\mathcal{D}_l$ and the $N_r^c - N_l^c$ samples in class c from $\mathcal{D}_u$ with the lowest loss ℓ_i.

Differently from previous works tackling synthetic noise [144], we find that the noise present in $\hat{\mathcal{D}}$ makes the clean sample retrieval using the loss ℓ_i during any particular epoch unstable and that the noise is class-unbalanced (see Table 5.1), making it more challenging. We therefore impose the selection of a class-balanced clean subset and choose to average the network losses over the last T training epochs. This results in a clean, trusted subset which limits the label noise bias introduced to the semi-supervised algorithm. Table 5.3 shows that the knowledge we bootstrap in $\mathcal{D}_r$ is not overly sensitive to N_r.

5.2.3 Semi-supervised learning

Unlike traditional learning from $\mathcal{D}_l$ and $\mathcal{D}_u$, ReLaB provides semi-supervised algorithms with a (larger) reliable labeled set $\mathcal{D}_r$ extended from the original (smaller) labeled set $\mathcal{D}_l$. The extension from $\mathcal{D}_l$ to $\mathcal{D}_r$ is done in a completely unsupervised manner and promotes a significant reduction of the error rates of SSL algorithms when few labels are given.

5.3 Experiments

5.3.1 Datasets and implementation details

We experiment with three image classification datasets: CIFAR-10 [99], CIFAR-100 [99], and mini-ImageNet [202]. We follow common practices for image retrieval [13, 154] and perform PCA whitening as well as L_2 normalization on the features v before applying diffusion. We construct the reliable set $\mathcal{D}_r$ by training for 60 epochs with a high learning rate (0.1) to prevent label noise memorization [8] and select the samples with the lowest loss per class at the end of the training. We average the per-sample loss over the last $T = 30$ epochs of training. For the semi-supervised learning experiments, we always use a standard WideResNet-28-2 [221] for fair comparison with related work. We combine our approach with state-of-the-art pseudo-labeling [9] and consistency regularization-based [19] semi-supervised methods to demonstrate the stability of ReLaB when applied to different semi-supervised strategies. We use the default configuration for pseudo-labeling[1] except for the network initialization, where we make use

[1] https://github.com/EricArazo/PseudoLabeling

of the Rotation unsupervised objective [61] and freeze all the layers up to the last convolutional block in a similar fashion to Rebufi et al. [158]. We find that this is necessary to preserve strong early features throughout the training. The network is warmed up on the labeled set for 200 epochs and then trained for 400 epochs on the whole dataset. For ReMixMatch[2] we train the network from scratch for 256 epochs. Experiments in Section 5.3.3 for the supervised alternatives on dealing with label noise [8, 227] follow the author's configurations, while cross-entropy and Mixup training in Table 5.4 is done for 150 epochs with an initial learning rate of 0.1 that we divide by 10 in epochs 80 and 130. The benchmark datasets and hyperparameters are the same as the ones used in related state-of-the-art literature [17, 9]. The algorithms we compare with in this section all achieved state-of-the-art results on the benchmark datasets at the time of their publication.

5.3.2 Importance of unsupervised representations quality for label propagation

Label propagation relies upon unsupervised representations extracted form the data, i.e. the quality of the propagation directly depends on these representations. We propose to explore different unsupervised learning alternatives to obtain these representations. Table 5.2, presents the label noise percentage of the extended labeled set $\hat{\mathcal{D}}$ in CIFAR-10 (100) formed after label propagation of the specified unsupervised representations with 1, 4 and 10 (4, 10 and 25) labeled samples per-class in $\mathcal{D}_l$. We select RotNet [61], NPID [212], UEL [126], AND [86] and iMix [107] as five recent unsupervised methods. We experiment training WideResNet-28-2 (WRN-28-2) [221], ResNet-18

[2] https://github.com/google-research/remixmatch

Table 5.2: Label noise percentage in $\hat{\mathcal{D}}$ using different amounts of labeled samples per class after label propagation using different unsupervised methods and network architectures. We train RotNet, Non Parametric Instance Discrimination (NPID), Unsupervised Embedding Learning (UEL), Anchor Neighboring Discovery (AND) and iMix. Lower is better.

		CIFAR-10			CIFAR-100		
		1	4	10	4	10	25
	WRN-28-2	67.90 ± 8.51	51.68 ± 3.03	50.09 ± 2.55	83.08 ± 0.52	76.31 ± 0.33	67.81 ± 0.15
RotNet [61]	RN-18	66.02 ± 5.98	53.58 ± 1.57	47.60 ± 3.51	80.83 ± 0.56	73.79 ± 0.42	65.58 ± 0.34
	RN-50	80.52 ± 30.08	77.58 ± 3.45	71.07 ± 1.05	80.75 ± 0.23	72.33 ± 0.15	62.78 ± 0.12
	WRN-28-2	68.72 ± 1.51	56.3 ± 2.42	51.35 ± 1.55	84.02 ± 0.30	76.91 ± 0.40	67.97 ± 0.13
NPID [212]	RN-18	59.34 ± 7.13	42.70 ± 2.32	37.14 ± 0.48	77.80 ± 0.55	69.54 ± 0.25	61.29 ± 0.67
	RN-50	59.44 ± 3.10	44.54 ± 2.32	38.13 ± 0.63	76.67 ± 0.58	68.54 ± 0.10	60.46 ± 0.16
	WRN-28-2	60.81 ± 6.41	45.84 ± 2.09	41.30 ± 2.00	79.21 ± 0.09	71.29 ± 0.39	62.89 ± 0.19
UEL [126]	RN-18	52.02 ± 7.24	34.51 ± 1.03	29.84 ± 0.78	71.90 ± 0.36	63.25 ± 0.41	56.51 ± 0.22
	RN-50	49.48 ± 7.66	32.81 ± 1.50	28.78 ± 1.08	69.62 ± 0.13	60.81 ± 0.48	54.08 ± 0.22
	WRN-28-2	61.35 ± 0.57	46.12 ± 4.07	40.78 ± 0.27	79.38 ± 0.37	71.65 ± 0.03	63.29 ± 0.38
AND [86]	RN-18	46.55 ± 5.64	28.82 ± 1.29	24.64 ± 1.44	67.48 ± 1.04	58.3 ± 0.26	51.47 ± 0.13
	RN-50	41.96 ± 8.74	24.34 ± 0.94	21.28 ± 0.75	66.25 ± 0.33	56.6 ± 0.52	46.31 ± 0.15
iMix[107]	WRN-28-2	53.75 ± 2.58	37.06 ± 2.40	31.27 ± 0.27	76.26 ± 0.60	64.92 ± 0.18	57.95 ± 0.45
+	RN-18	46.25 ± 6.11	18.55 ± 1.81	14.51 ± 2.35	49.74 ± 1.20	42.90 ± 0.39	39.17 ± 0.26
N-pairs	RN-50	**38.14 ± 8.34**	**16.93 ± 1.73**	**13.72 ± 1.70**	**45.49 ± 1.04**	**39.41 ± 0.08**	**35.75 ± 0.26**

(RN-18) and ResNet-50 (RN-50) [92] architectures. All the unsupervised methods are trained using the recommended configuration. We report average noise percentage and standard deviation for 3 different labeled subset $\mathcal{D}_l$. We confirm that the architecture has a key impact on the label noise percentage, which agrees with previous observations on the quality benefits of unsupervised features from larger architectures [97, 107]. We find that using diffusion on features learned using the iMix algorithm promotes the lowest amount of noise and adopt it together with a ResNet-50 in the subsequent experiments.

5.3.3 How accurately must noisy labels be detected?

We analyze the importance of the selected number of samples N_r over the label noise percentage in the extended reliable subset $\mathcal{D}_r$ and semi-supervised

Table 5.3: Sensitivity of semi-supervised methods to different amounts of bootstrapped samples per class ($\frac{N_r}{c}$) considering an initial 4 labeled samples per class ($\frac{N_l}{c} = 4$). We report label noise percentage in $\mathcal{D}_r$ and final error rates after semi-supervised training.

	CIFAR-10		CIFAR-100	
$\frac{N_r}{C}$	Noise (%)	SSL error	Noise (%)	SSL error
25	**0.40**	12.12	**25.48**	51.90
50	0.60	9.18	30.20	51.43
75	1.07	**8.76**	33.51	**50.65**
100	1.30	8.79	35.69	51.14

Table 5.4: Learning from $\hat{\mathcal{D}}$ constructed from 4 labeled samples per class on CIFAR-10 ($N_l = 40$) and CIFAR-100 ($N_l = 400$). We train a naive cross-entropy objective (CE), Mixup (M), Dynamic Bootstrapping (DB), Early Learning Regualrization (ELR). Error rates

	CIFAR-10	CIFAR-100
CE	22.64	59.88
M [227]	21.27	57.92
DB [8]	14.84	55.07
ELR [119]	17.39	47.95
Ret. score + PL [9]	17.55	54.19
ReLaB + PL [9]	12.38	53.58
ReLaB + RMM [19]	**6.68**	**43.53**

performance (using RMM [19]). Table 5.3 shows how a balance has to be found between a sufficient amount of bootstrapped samples and a low noise ratio. Increasing the number of samples in $\mathcal{D}_r$ is beneficial up to 100 samples per class, where adding more does not compensate the higher noise percentage. Based on this experiment and the typical amounts of labeled samples needed to perform successful SSL [9, 17, 81, 190], we choose a conservative $N_r = 500$ (4000) for CIFAR-10 (100) for further experiments.

Since $\hat{\mathcal{D}}$ is corrupted with label noise, it is reasonable to expect that supervised alternatives on dealing with label noise [8, 227] could help combat this label noise. Table 5.4 compares our proposed approach against train-

Table 5.5: ReLaB for semi-supervised learning on CIFAR-10 and CIFAR-100 with very limited amounts of labeled data. Error rates. We mark with † the methods we run ourselves. Other results are from [178] or [207]. We compare against the π-model, mean-teachers (MT), pseudo-labeling (PL), MixMatch (MM), Unsupervised Data Augmentation (UDA), ReMixMatch (RMM) and Ensemble of Auto- Encoding Transformations (EnAET) Bold denotes best.

	CIFAR-10			
Labeled samples	10	40	100	250
π-model [157]	-	-	-	54.26 ± 3.97
MT [190]	-	-	-	32.32 ± 2.30
PL [9]†	55.61 ± 5.28	29.65 ± 5.71	12.83 ± 0.68	12.00 ± 0.32
MM [17]	-	47.54 ± 11.50	-	11.05 ± 0.86
UDA [214]	-	29.05 ± 5.93	-	8.82 ± 1.08
RMM [19]†	58.80 ± 1.98	31.36 ± 4.37	22.56 ± 2.58	7.80 ± 0.83
EnAET [207]	-	-	9.35	7.60 ± 0.34
ReLaB + PL†	29.89 ± 3.64	12.38 ± 0.78	11.38 ± 0.64	10.68 ± 0.66
ReLaB + RMM†	$\mathbf{22.34 \pm 4.92}$	$\mathbf{8.23 \pm 1.38}$	$\mathbf{6.89 \pm 0.18}$	$\mathbf{6.71 \pm 0.20}$

	CIFAR-100			
Labeled samples	100	400	1000	2500
π-model [157]	-	-	-	57.25 ± 0.48
MT [190]	-	-	-	53.91 ± 0.57
PL [9]†	88.23 ± 0.32	67.57 ± 0.58	55.20 ± 0.69	45.42 ± 0.68
MM [17]	-	67.61 ± 1.32	-	39.94 ± 0.37
RMM [19]†	81.18 ± 2.36	57.44 ± 2.53	44.11 ± 1.51	36.66 ± 0.33
EnAET [207]	-	-	58.73	-
ReLaB + PL†	68.04 ± 2.52	53.58 ± 1.20	48.79 ± 0.82	43.84 ± 0.72
ReLaB + RMM†	$\mathbf{62.02 \pm 2.77}$	$\mathbf{44.09 \pm 0.51}$	$\mathbf{39.58 \pm 0.70}$	$\mathbf{35.19 \pm 0.74}$

ing on $\hat{\mathcal{D}}$ with standard cross-entropy (CE) and label noise robust methods such as Mixup (M) [227], the Dynamic Bootstrapping (DB) loss correction method [8] and the Early Regularization (ELR) strategy [119]. We also report using the retrieval score (Ret. score) from the label propagation ($\max_c F_{ic}$ in eq. 5.2) instead of ReLaB for selecting the trusted subset. In both CIFAR-10 and CIFAR-100, ReLaB + RMM outperforms supervised alternatives.

5.3.4 Semi-supervised learning with ReLaB

Table 5.5 presents the benefits of ReLaB for semi-supervised learning, showing great improvements for both PL [9] and ReMixMatch (RMM) [19] when

Table 5.6: Effect of ReLaB on mini-ImageNet with very limited amounts of labeled data and $N_r = 4000$. Error rates.

Labeled samples	100	400	1000	2500
PL [9]	90.89 ± 0.62	85.00 ± 0.94	75.47 ± 0.52	55.10 ± 1.52
ReLaB + PL	$\mathbf{76.25 \pm 0.80}$	$\mathbf{66.66 \pm 0.54}$	$\mathbf{60.82 \pm 1.04}$	$\mathbf{52.39 \pm 1.03}$

paired with ReLaB. Our focus is on very low levels of labeled samples as semi-supervised methods [19] already achieve very good performance with larger numbers. We further study the 1 sample per class scenario in Section 5.3.5. Table 5.6 demonstrates the scalability of our approach to higher resolution images by evaluating ReLaB + PL [9] on mini-ImageNet [202]. Due to GPU memory constrains, we use ResNet-18 instead of ResNet-50 to train iMix with an acceptable batch size for the mini-ImageNet experiments.

5.3.5 Very low levels of labeled samples

The high standard deviation using 1 sample per class ($N_l = 10$) in CIFAR-10 (Table 5.5) motivates the proposal of a more reasonable method to compare against other approaches. To this end, Sohn et al. [178] proposed 8 different labeled subsets for 1 sample per class in CIFAR-10, ordered from more representative to less representative, we reduce the experiments to 3 subsets: the most representative, the least representative, and one in the middle. Figure 5.2 shows the selected subsets; the exact sample ids are available together with our code for easy reproduction.

Table 5.7 reports the performance for each subset and compares against FixMatch [178] and ReMixMatch [19]. Note that the results obtained for the less representative samples reflect the results that can be expected on average when drawing labeled samples randomly. In the case of the not

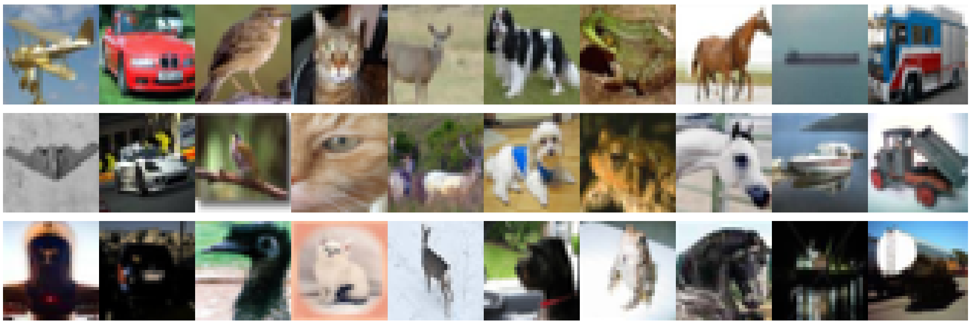

Figure 5.2: Labeled samples used for the 1 sample per class study on CIFAR-10 and taken from [178], ordered from top to bottom from most representative to least representative.

Table 5.7: Error rates for 1 sample per class on CIFAR-10 with different labeled sets. We run all the methods ourselves except for FixMatch [178]. Key: MR (Most Representative), LR (Less Representative), NR (Not Representative).

	MR	LR	NR
ReMixMatch [19]	50.62	62.57	90.00
FixMatch [178]	22.00	35.00	90.00
ReLaB + PL	19.86	32.38	79.9
ReLaB + RMM	**8.46**	**21.75**	**78.25**

representative subset, ReLaB enables the semi-supervised learning algorithms to converge better than a random guess. We find that for CIFAR-100 and mini-ImageNet, runs across different initial labeled samples are more consistent and a comparison to other methods can be made even when drawing the labeled samples at random.

5.3.6 Ablation study over the importance of data augmentation for a clean subset selection

We perform an ablation study on the importance of data augmentation for selecting a reliable subset. Table 5.8 reports the noise ratio for the bottom 20% of the samples with the lowest average loss over the last 30 epochs.

Table 5.8: Influence of data augmentation (DA) on the noise ratio for the 20% lower loss samples at the end of the training. We report results for CIFAR-10 and CIFAR-100 with 400 and 4000 labeled samples respectively. Lower is better

Dataset	CIFAR-10	CIFAR-100
# labeled samples	40	400
No DA	2.76	**19.11**
Weak DA	2.66	19.93
Weak DA + Color Jitter	**2.07**	20.26
Weak DA + Mixup [227]	2.66	20.13

We train with a WideResNet-28-2 from scratch for 60 epochs with a fixed learning rate of 0.1 to avoid fitting the noise. Weak data augmentations (DA) denotes vertical and horizontal random flipping as well as random cropping. All experiments are run on the same noisy set $\hat{\mathcal{D}}$, obtained from propagating 4 samples per class for CIFAR-10 and CIFAR-100. We find that noise ratios obtained are very stable independently of the augmentation strategy so we choose to not augment the samples during the clean subset selection.

5.3.7 ReMixMatch training

We report in Table 5.9 the error rates for ReMixMatch, with and without our proposed ReLaB method, when training for 256 epochs instead of the original 1024 [19]. Although longer training is always beneficial, we observe convergence to a reasonable performance in 256 epochs. We adopt the 256 configuration as it substantially reduces training time.

5.3.8 Ablation study for pseudo-labeling

We study the effect on the pseudo-labeling algorithm in [9] when using an unsupervised initialization with RotNet [61] and freezing all the layers up

Table 5.9: Error rate of short and long training of the RMM algorithm [19]. We report mean and standard deviation over 3 runs.

Dataset	CIFAR-10	CIFAR-10	CIFAR-100
# labeled samples	250	40	400
ReLaB	No	Yes	Yes
RMM - 256	7.8 ± 0.83	10.04 ± 4.58	48.59 ± 0.7
RMM - 1024	6.24 ± 0.34	9.04 ± 4.37	47.24 ± 0.68

Table 5.10: Ablation study on the error rate of pseudo-labeling (PL) algorithm in [9] combined with ReLaB when using unsupervised initialization and layer freezing (LF).

Dataset	CIFAR-10	CIFAR-100
# labeled samples	40	400
ReLaB + PL [9]	22.12	58.17
ReLaB + PL (RotNet [61])	15.72	59.04
ReLaB + PL (RotNet [61] + LF)	$\mathbf{14.21}$	$\mathbf{57.09}$

to the last convolutional block to avoid fitting label noise of the reliable extended set $\mathcal{D}_r$. Unsupervised initialization and early layers freezing is also adopted in [158] to improve pseudo-labeling. We show in Table 5.10 that both strategies contribute to better pseudo-labeling performance.

5.3.9 Visualization of the bootstrapped samples

Figure 5.3 displays the capacity of our reliable sample selection to select an extended clean subset for the semi-supervised algorithm. The first row displays the (initial) seed samples; the middle row display a random subset of the samples labeled using label propagation on unsupervised features; the last row displays the reliable samples we select to extend the label set. Images with a red border have a noisy label. The label noise is reduced in the reliable extended set. The figure is best viewed on a computer.

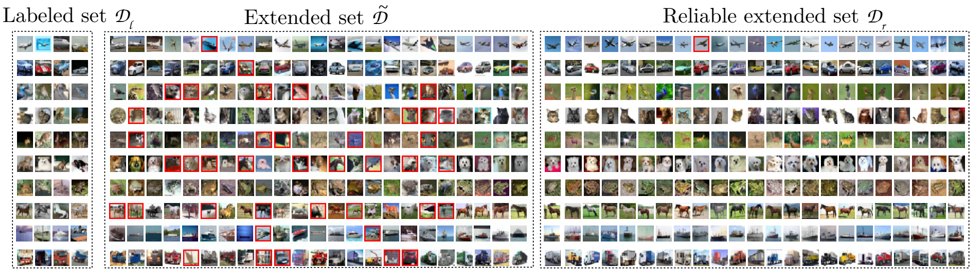
Labeled set $\mathcal{D}_l$
Extended set $\tilde{\mathcal{D}}$
Reliable extended set $\mathcal{D}_r$

Figure 5.3: Qualitative example of label propagation and reliable sample selection in CIFAR-10 with four seed samples per class. Best viewed on a computer.

5.4 Conclusion

This chapter aims to answer research question three: *“Can unsupervised features be used as a medium to propagate labels in a semi-supervised scenario when few labels are available?”*. The algorithm proposed in this chapter leverages methods from different vision tasks (image retrieval, unsupervised feature learning, label noise for image classification) to propose a bootstrapping of additional labeled samples using unsupervised features, which can in turn be used to enhance any semi-supervised learning algorithm. We demonstrate the direct impact of better unsupervised features for the performance of ReLaB and the relevance of our reliable sample selection. Using the extended amount of supervision of ReLaB’s reliable set, we enable semi-supervised algorithms to reach remarkable and stable accuracies with very few labeled samples on standard datasets. The extremely low levels of labeled samples we consider in this chapter (< 25 per class) addresses a gap in the semi-supervised literature, which otherwise perform on par with supervised learning for moderate levels of labeled samples (> 25 per class), see Table 5.5. Direct applications of ReLaB would include scenarios where the annotation of images is very time consuming or requiring expert annotators. In the case of this thesis, an application of interest for semi-supervised learning is to estimate the biomass composition of herbage from images which is a regression task. Chapter 6 will introduce the grass biomass prediction problem and propose a specially designed semi-supervised solution and Chapter 7 will study if existing semi-supervised solutions for image classification can be translated to this specialist regression task where ground-truth acquisition is destructive.

Chapter 6

Semi-supervised dry herbage mass estimation from noisy automatic labels and synthetic images

This chapter presents a semi-supervised approach to predict herbage characteristics from images. The solution proposed is not directly inspired from the semi-supervised literature for image classification but is very specific to the problem of predicting grass composition from canopy views. The chapter will detail how unlabeled images were used together with few labeled images to train a semi-supervised solution that improves the prediction accuracy over using the few labeled samples alone. Section 6.1 motivates the need for biomass composition estimation in grasslands, why computer vision holds great potential for a quick visual composition estimation and exposes the research contributions of the chapter. Section 6.2 presents a detailed description of the proposed algorithm. Section 6.3 contains implementation details, an ablation study, andcomparison with the state-of-the-art. Section 6.4 concludes the research carried out in the chapter. The research that emanated from

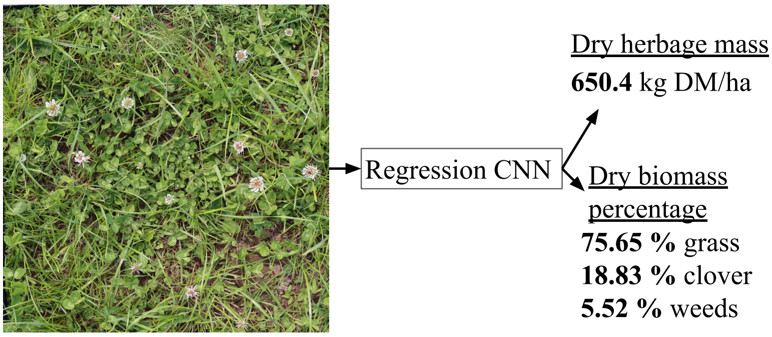

Figure 6.1: Overview of the dry herbage mass prediction task

this work was published at the 7th workshop on Computer Vision in Plant Phenotyping and Agriculture (CVPPA) workshop at the 2021 IEEE/CVF International Conference on Computer Vision (ICCV).

6.1 Motivations

Nitrogen fertilization has proven to be efficient in enhancing grass quantity and quality, yet over-fertilization has detrimental effects on biodiversity and on the environment in general [6, 91, 134]. In this context, clover proves to be an important ally to the farmer for two reasons. First, clover naturally captures widely available nitrogen from the atmosphere and renders it available in the soil for the grass to use [177, 142]. Second, having proper amounts of clover in the feed has been shown to increase cow appetite, which in turn translates to higher milk production [132, 53]. Monitoring clover content in the herbage then becomes an important aspect of milk production and regular herbage biomass probing is performed by humans to ensure a proper grass to clover balance. The herbage probing process involves cutting a sample from the field, drying it in lab before manually separating each component of the herbage by hand [53] making it a long and expensive process.

Species phenotyping proposes a direct application of computer vision where a canopy view of the objects is passed to an algorithm tasked with a computer vision problem. Some examples of these tasks include semantic segmentation [174, 71, 129], object counting [95, 12], classification [42, 62, 130], object detection [89, 164] and regression [132, 135]. The principal limitation when applying deep learning approaches to species phenotyping remains the large amount of annotated data required. Lower supervision alternatives using semi-supervised or unsupervised approaches can lower the annotation burden and enable a stronger convergence than using a small number of annotated images alone. In the case of grass/clover biomass estimation this is even more important, as the annotation process is destructive. To accurately measure biomass the region of interest has to be cut, separated, and weighed in a laboratory whereas the collection of un-annotated images is fast and simple.

In this chapter, we use a large collection of unlabeled images together with a small annotated subset to improve the accuracy of a dry herbage mass predicting convolutional neural network (CNN, see Figure 6.1). We first learn a weakly-supervised semantic segmentation network on synthetic images to estimate the species density in the herbage. We then use the segmentation masks to generate automatic biomass labels for the unlabeled images using a simple regression algorithm. Finally, we train a convolutional neural network on a mix of the automatically labeled data and a small number of manually labeled examples to improve the regression accuracy over training on the small number of manually labeled examples alone. We construct our algorithm on an Irish dry herbage mass dataset [75] and validate our results on a publicly available dry biomass dataset [174] collected in Denmark.

6.2 Biomass prediction in grass-clover pastures

This section introduces the semi-supervised learning problem of dry biomass estimation of grass-clover pastures, the datasets used, the synthetic image generation process, the automatic labelling pipeline, and our automatic label robust biomass regression algorithm.

6.2.1 Semi-supervised biomass estimation in grass-clover pastures

We consider here a semi-supervised regression problem with $X_L = \{x_i\}_{i=1}^{L}$ labeled canopy images of grass and clover, and their corresponding label assignment $Y = \{y_i\}_{i=1}^{L}, Y \in \mathbb{R}^S$ where S is the number of species to predict in the herbage. The small labeled set is complemented by a large set of unlabeled images $X_U = \{x_i\}_{i=1}^{D}$ with no corresponding labels and $|X_U| \gg |X_L|$. We note the complete dataset used to train the network $X = X_L \cup X_U$. This chapter aims to solve the dry biomass prediction problem from images using a convolutional neural network $\Phi : X \rightarrow Y$ using unlabeled images to the improve the regression accuracy.

6.2.2 Grass clover dry biomass datasets

We consider two different dry biomass prediction datasets, both centered around grass and clover biomass prediction. The first one is the publicly available GrassClover dataset [175]. This dataset is composed of 157 annotated images (to be divided between training set and validation set) and 31.600 unlabeled images. The image acquisition was carried out in Danish fields between 2017 and 2018 using for the most part an ATV mounted camera. The

Research published in the Computer Vision for Plant Phenotyping and Agriculture Workshop (CVPPA) at ICCV 2021 ground-truth collected is composed of the dry biomass percentages for the grass, white clover, red clover, total clover and weeds. The second dataset is the Irish clover dataset [75], which is composed of 424 training images, 104 held out test images, and 594 unlabeled images. The images were captured in the south of Ireland in the Summer of 2020 using a camera mounted on a tripod. The ground-truth collected is composed of the dry biomass percentages for grass, total clover and weeds ($\%$), the herbage height (cm), and the herbage dry matter per ha (kg DM/ha). The Irish dataset additionally proposes images captured using handheld phone devices where some quadrat captured using the high resolution camera are also captured using a phone. Each image in the validation set is available in either camera or phone format. 295 unlabeled phone images are also supplied.

6.2.3 Herbage height aware semantic segmentation on synthetic images

The task we aim to solve in this section is to first predict a semantic segmentation of the herbage into grass, clover (possibly red-white), and weeds; and second, a herbage height map. Since human annotation of ground truth for semantic segmentation can take up to several hours per image [118] and since a pixel specific herbage height is difficult to estimate in practice, we propose (similar to [174]) to train our semantic segmentation network Ψ on a synthetically generated dataset $\tilde{X}$. We generate the synthetic semantic segmentation images together with their 100$\%$ pixel-accurate synthetic segmentation ground truth using manually cropped out elements from the unlabeled images. In accordance to the low supervision scope of this chapter, we only crop out 78 samples (see Figure 6.2) and collect 8 bare soil images

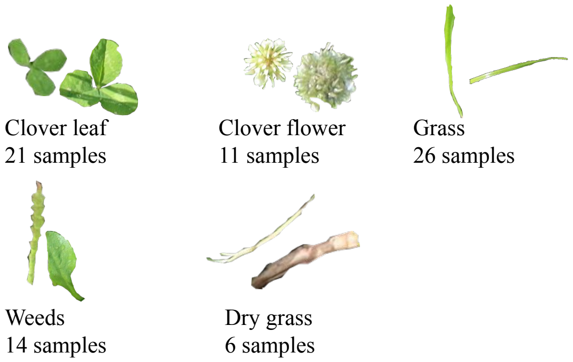

Figure 6.2: Cropped out samples for every species

to paste elements onto. The bare soil images are collected at the same site and using the same equipment as Hennessey et al. [75] during the Summer of 2021.

To produce images similar to the real images we aim to make predictions for, we respect the species ratio in images by enforcing the probability of a species to be pasted according to the observed average dry biomass distribution in the training dataset: 90% grass, 7% clover, 3% weeds. We draw the probability of each species to be pasted from a 3 component Dirichlet distribution with parameters $(9, 2, 1)$ for (grass, clover, weeds). Once the species has been decided, we randomly draw a sample for this category and apply a series of transformation to increase the diversity of the synthetic images. The transformations include: (uniform) random rotation $(\pm 180°)$, random Gaussian blur $(\text{radius} \in [0, 5])$, random brightness change $[0.6, 1]$, and random resizing $(50 - 150\%)$. Finally, we select a random center location to paste the sample on the background images as well as a mask of the sample's label on the ground truth map. We additionally approximate the herbage height in the synthetic images as the sum of the total number of successive elements pasted on a pixel. In the rest of the chapter, this approximation made on synthetic images will be referred to as herbage height. For example, if three samples

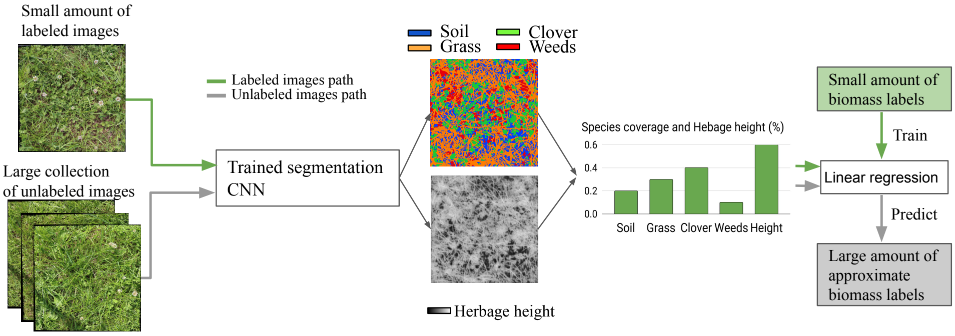
Small amount of labeled images
Large collection of unlabeled images
Labeled images path
Unlabeled images path
Trained segmentation CNN
Soil
Grass
Clover
Weeds
Herbage height
Species coverage and Hebage height (%)
0.6
0.4
0.2
0.0
Soil
Grass
Clover
Weeds
Height
Small amount of biomass labels
Train
Linear regression
Predict
Large amount of approximate biomass labels

Figure 6.3: Automatic labeling from semantic segmentation

have been pasted at the same pixel (clover on top of grass on top of clover), we define the un-normalized herbage height as 3 for the given pixel. Once the synthetic dataset has been fully generated, we compute the 75th percentile of the herbage height for every pixel in all generated images (allowing us to filter outliers) and use this value to clip overly high herbage height numbers and produce a normalized herbage height between 0 and 1 for every pixel in every synthetic image. The normalized herbage height becomes the ground truth target for the segmentation network. Additionally, we found that the quality of the segmentation learnt by Ψ is best when the number of elements to paste is in $[400, 800]$ per image (randomly varied across images); beyond this the synthetic images become overly cluttered. Images are generated at a 2000×2000 resolution. The RGB images are stored in the JPEG format, the grayscale ground truth maps are stored as PNG images, and the herbage height matrix is stored as a compressed numpy array. Figure 6.3 illustrates the automatic labeling pipeline.

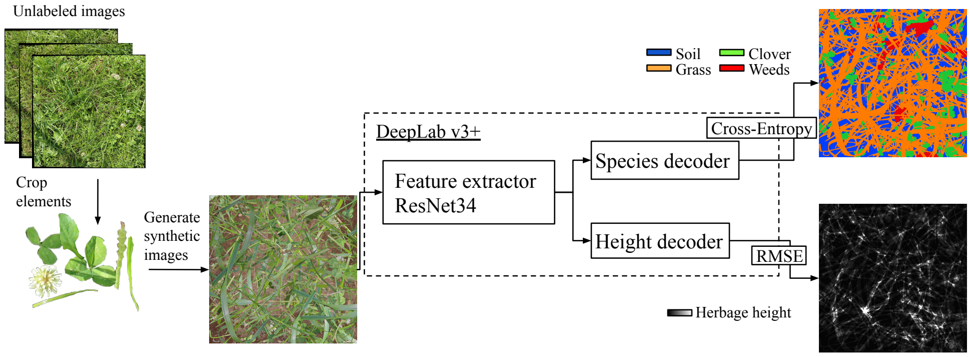

Figure 6.4: Herbage height aware semantic segmentation on synthetic images

6.2.4 Generating synthetic images suitable for herbage mass estimation

To concurrently solve the tasks of semantically segmenting the herbage images and estimating the herbage height for every pixel in the images, we propose a herbage height aware semantic segmentation network Ψ consisting of a single feature extractor coupled with two decoder branches (see Figure 6.4). We concurrently train the species segmentation branch using a pixel-level cross-entropy loss:

$$l_{\text{species}} = -\sum_{i=1}^{C} \hat{y}_i \log(s_i),$$

where $S = \{s_i\}_{c=1}^{C}$ is the softmaxed prediction of the network and $\hat{Y} = \{\hat{y}\}_{i=1}^{C}$ are the synthetic segmentation labels. The herbage height branch is trained using a root mean square error (RMSE) loss:

$$l_{\text{height}} = \sqrt{\frac{1}{P}\sum_{p=1}^{P}\left(\hat{h} - h\right)^2},$$

where P is the total amount of pixels in the images, h is the ground truth synthetic height label, and $\hat{h}$ is the network prediction (sigmoid). The total training loss of the segmentation network Ψ is $l = l_{\text{specices}} + l_{\text{height}}$.

6.2.5 Automatic label prediction from species density estimations

The herbage height aware semantic segmentation network Ψ allows us to reduce the complexity of the biomass prediction problem by simplifying the input domain from high resolution real RGB images to the surface area occupied by each species in the canopy as well as an estimated herbage height map. From there, we compute the relative area occupied by each species in the canopy (in $\%$) and the predicted herbage height over each image and train a simple ridge regression algorithm using the small number of labels, Y, to predict approximate labels for X_U. This intermediate task allows us to generate accurate automatic labels for X_U even if the number of images in X_L is very limited.

6.2.6 Regression on automatic labels with a trusted subset

Although the biomass information can be directly predicted using the automatic annotation process (as done in Skovsen et al. [174]), we propose to attempt to decrease the regression error further by solving the regression problem directly from the RGB images using a convolutional neural network, Φ, and both human-labeled and automatically labeled image datasets: X_L coupled with ground truth labels Y (the trusted set) and X_U coupled with approximate labels $\hat{Y}$ (the automatically labeled set). Φ is trained to predict the biomass composition ($\%$) and the dry herbage mass (kg DM/ha) from RGB images directly; the automatic images are only used in Ψ to help predict the automatic labels $\hat{Y}$ for unlabeled images in X_U. To ensure that Φ will not over-fit to incorrect approximate labels, we use three mechanisms. First, we over-sample the trusted data to ensure that a fixed percentage will always be presented to the network in every mini-batch ($\frac{3}{4}$ approximate labels, $\frac{1}{4}$ trusted labels). Second, we use a label perturbation strategy where we randomly perturb the automatic labels to avoid over-fitting incorrect targets, and to avoid penalizing the network for making a prediction slightly different than the incorrect prediction. In practice, we randomly perturb the label in the interval of $\pm$ two times the observed RMSE of the automatic labels on the validation set. Finally, we find that adding vertical flipping and randomly grayscaling to the input images to be interesting augmentations that preserve the full herbage information of the image and help further decrease validation error.

6.3 Experiments

6.3.1 Training details

We use two different neural networks to solve two distinct tasks. For the semantic segmentation network Ψ, we use a state-of-the-art architecture: DeepLabV3+ [28] where we duplicate the decoder to create the herbage height branch. Ψ is trained on 800 synthetic images and uses 200 synthetic images for validation. We use a ResNet34 [92] as the feature extractor, initialized on ImageNet [100], and with an output stride of 16 for both training and testing. We use the “poly” lr schedule [28] starting at 0.007, a batch size of 4, and train for 60 epochs. For the base data augmentation we resize images to 1024 on the short size, randomly crop a 1024×1024 square, randomly flip horizontally, and normalize the images.

For the regression network Φ, we use a ResNet18 network [221] pretrained on ImageNet to solve the regression problem from RGB images directly. We train for 100 epochs, starting with a learning rate of 0.03 dividing it by 2 at epochs 50 and 80. We use the same base data augmentation as for Ψ but with a resolution lowered to 512×512. For the strong(er) data augmentation, we add random vertical flipping and random grayscaling ($p = 0.2$). We train with a batch size of 12.

We use the Irish dataset [75] in its low supervision configuration (52 images are used for training, 104 for validation and 372 for testing) for our exploratory studies, and generate 1000 synthetic images to train Ψ according to the process described in Section 6.2.4. We validate our results on the GrassClover dataset [174] and use the full 152 fully annotated biomass images, dividing them into 100 for training and 52 for validation; we use the 174 images withheld for the CodaLab [1] for testing. We make use of 800 randomly selected synthetic images out of the 8000 generated by the authors for Ψ, keeping 200 extra images for validation. We do not train the herbage height branch on the GrassClover dataset.

To evaluate the performance of the algorithms, we report the RMSE when predicting the dry biomass species percentage for both the Irish and GrassClover datasets. For the Irish dataset, we additionally report the RMSE of the global herbage mass prediction (HRMSE, kg DM/ha), the herbage relative absolute error $l_{\text{relative}} = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y_i}|}{y_i}$ (HRAE, in $\%$) and the HRMSE specific to each species (kg DM/ha).

[1] https://competitions.codalab.org/competitions/21122

Table 6.1: Importance of data augmentation and batch normalization tuning when training on synthetic images.

	HRMSE						RMSE			
	Total	Grass	Clover	Weeds	Avg.	HRAE	Grass	Clover	Weeds	Avg.
Simple DA	357.35	328.66	55.74	26.75	137.05	35.26	8.11	6.87	3.22	6.07
+ ColorJitter	319.92	289.32	60.81	31.40	127.18	35.46	8.63	7.68	3.55	6.62
+ BN tuning	284.60	258.34	51.92	27.05	112.44	31.79	6.49	4.94	3.24	4.89

6.3.2 Semantic segmentation on synthetic images

To encourage Ψ to learn robust features that will generalize to unseen real images, we augment the synthetic images using color jittering and Gaussian blur. Furthermore, once the network has converged on the synthetic dataset and before predicting on the real images, we perform batch normalization tuning which is a common domain adaptation strategy [117] on the real images. An ablation study on the importance of the data augmentation and batch normalization tuning is given in Table 6.1, where we use the best performing regression algorithm from 6.3.3.

6.3.3 Regression from species coverage

We compare different sets of simple features to extract from the segmentation masks as well as the importance of the herbage height prediction when estimating the dry herbage mass. For features directly related to the dry biomass percentages, we compare averaging the most confident prediction for every pixel only (hard label, HL), averaging the full softmax prediction at each pixel (soft label, SL), or using the two sets of features jointly (HL+SL). In the regression model each feature is the average of the observations over the whole image: 4 features (soil %, grass %, clover %, weeds %) for HL or SL (8 for HL+SL), and 1 feature for the herbage height.

Table 6.2: Ablation study for predicting approximate labels. We report the biomass prediction errors on a heldout validation set. **HL**: hard labels, **SL**: soft labels, **H**: herbage height

	HRMSE						RMSE			
	Total	Grass	Clover	Weeds	Avg.	HRAE	Grass	Clover	Weeds	Avg.
HL	351.54	332.88	51.34	28.29	137.50	41.61	6.82	6.20	3.25	5.42
SL	310.68	279.98	57.48	28.15	121.87	34.18	7.61	5.20	3.24	5.35
HL + SL	315.20	288.52	53.37	28.11	123.33	34.33	6.49	4.91	3.23	4.88
HL + SL + H	284.60	258.34	51.92	27.05	112.44	31.79	6.49	4.94	3.24	4.89

We fit a least squares L_2 regularized (ridge) regression algorithm to all features with a regularization factor of 1, and train on the small subset of annotated images before evaluating on the validation set (Table 6.2). First, we report the RMSE error of the total herbage mass error (kg DM/ha), as well as the detailed grass/clover/weed herbage mass estimation (kg DM/ha). Second, we report the relative RMSE for the total herbage mass ($\%$) and the RMSE for the relative dry biomass estimation ($\%$) for the grass/clover/weeds. We notice that using SL is better than HL when predicting the herbage mass, demonstrating the interest of capturing the full softmax information over the max prediction only. We believe that the information contained in the soft label carries predictive information as to what the network expect to be present in the pixel hidden under the grass canopy where adjacent elements that become hidden under the current pixel are recognised. The information carried by SL is complementary to HL and we observe good improvements in terms of dry biomass percentage RMSE when the two sets of features are coupled. When adding the information about the herbage height, a decrease in HRMSE error is observed, validating the importance of the herbage height module in the segmentation architecture.

6.3.4 Biomass prediction using automatic labels and a trusted subset

We use the automatic labels to enhance the generalization of the regression CNN Φ in order to improve over the linear regression from the predictions of Ψ, especially in terms of herbage mass prediction. Table 6.3 reports the ablation study showing how the additional mechanisms we introduce allow us to be robust to the approximate automatic regression labels. The reported metrics are described in Section 6.3.1. We also compare the performance of the regression network against the linear regression from the prediction of Ψ.

6.3.5 Where are errors made?

We propose to study the type of compositions that are the hardest to predict for our algorithm. To do so, we plot predicted error rates against ground-truth values for herbage mass prediction and composition error in Figure 6.5. We plot the average biomass composition RMSE and the HRAE against the dry herbage mass ground-truth on the validation set and display image examples where the biggest errors are made. We observe that the highest errors are predicted on samples with low quantities of herbage in them (500kg DM/ha and less). Future data collection should focus on adding more training examples for this difficult category of images.

6.3.6 Transferability to phone images

We aim to test how well the knowledge learned by our algorithm transfers to a handheld phone images whose means of capture are much less normalized than high resolution camera images mounted on tripods. We study in Ta-

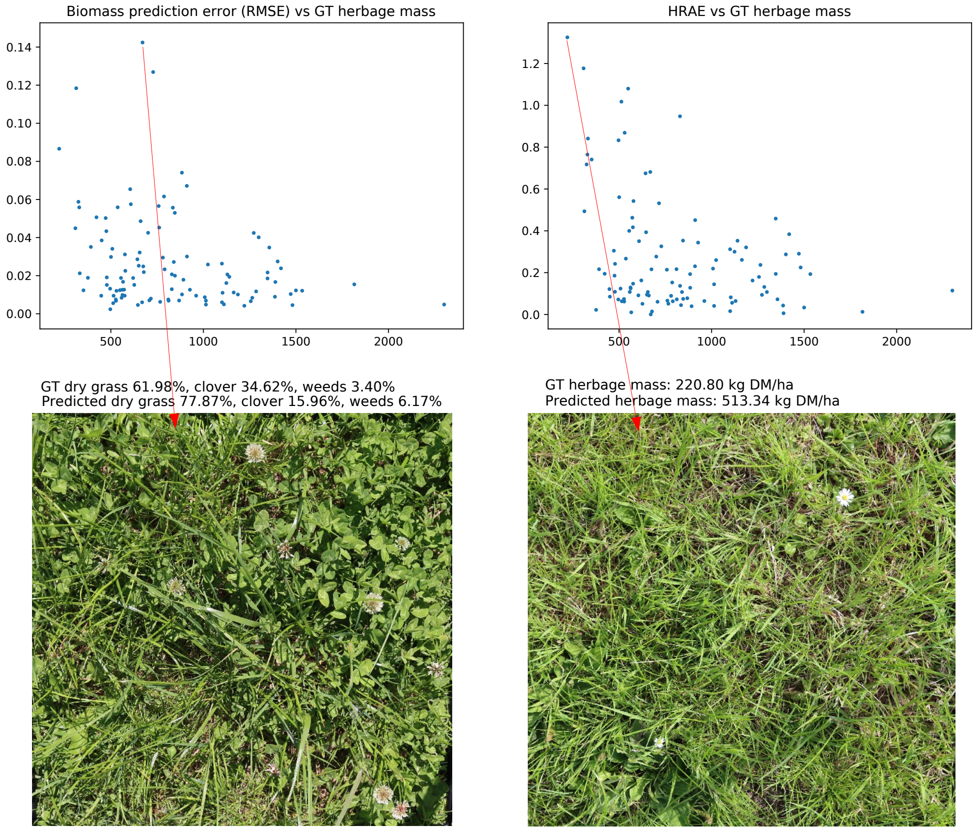

Figure 6.5: Error rates vs herbage quantity

ble 6.4 multiple scenario including: using the linear regression from semantic segmentation (LR); training Φ on labeled camera images only (trusted, T); training Φ on both trusted and automatically labeled camera images (T+A). We training Φ, we use the camera validation set for early stopping and test on the phone validation set. We observe that adding the automatically labeled data is important to reduce the error rate on the phone images. Although the error rates increase when moving from camera to phone images, Φ generalizes well to the phone data when automatically labeled images are introduced. We attempted to add unlabeled phone images when training Φ by automatically labeling them using Ψ and the linear regression but observe bad results because the automatic annotation regression fails to generalize well to phone

Table 6.3: Ablation study on training with approximate labels. We report results on the validation set using the linear regression baseline **LR** or training on the trusted data only **T**, the automatic data only **A**, or combinations of both **T+A**

	HRMSE					RMSE			
	Total	Grass	Clover	Weeds	HRAE	Grass	Clover	Weeds	Avg.
LR	284.60	258.34	51.92	27.05	31.79	6.49	4.94	3.24	4.89
T	249.48	253.63	45.62	32.67	21.67	6.28	5.07	3.94	5.10
A	258.00	239.81	46.51	27.74	23.48	5.72	5.20	3.29	4.74
T + A	245.04	233.34	34.94	26.32	21.60	4.70	4.45	3.17	4.11
+ random GS	234.25	217.55	37.57	27.72	21.55	4.66	4.47	3.27	4.13
+ trusted oversampling	232.08	220.09	35.93	26.34	21.36	4.33	4.17	3.15	3.88
+ random perturbation	229.93	216.23	35.79	26.05	19.96	4.22	4.21	3.10	3.84

data (first row in Table 6.4). This is mostly due to the segmentation algorithm Ψ failing, possibly because the phone images are blurrier.

6.3.7 Comparison against other works on the GrassClover dataset

We compare the improvements of our approach on the publicly released GrassClover dataset [174]. The target metrics for this dataset are limited to the dry biomass percentages, for which we report RMSE errors. Table 6.5 reports the performance of our algorithm with and without automatic labels on the test set available on the CodaLab challenge [2] and compares against the best available results. We report a lower RMSE on average than the methods we compare against and show that our algorithm is capable of using unlabeled images to reduce the biomass estimation error for every species over training on the small trusted subset alone.

[2] https://competitions.codalab.org/competitions/21122

Table 6.4: Generalization to phone images

	HRMSE					RMSE			
	Total	Grass	Clover	Weeds	HRAE	Grass	Clover	Weeds	Avg.
LR	370.50	348.71	93.04	36.39	0.31	12.01	10.21	4.31	8.84
T	368.76	366.75	59.42	33.45	0.41	10.75	8.80	4.15	8.37
T+A	280.51	268.66	45.22	34.51	0.32	6.59	5.75	3.45	5.26

Table 6.5: Results on the GrassClover test set (RMSE).

		Clover				
	Grass	Total	White	Red	Weeds	Avg.
Skovsen et al. [174]	9.05	9.91	9.51	6.68	6.50	8.33
Naranayan et al. [135]	8.64	8.73	8.16	10.11	6.95	8.52
Trusted data	10.28	10.32	9.24	9.54	7.37	9.35
+ Automatic data	8.78	8.35	7.72	7.35	7.17	7.87

6.4 Conclusion

This chapter aims to provide answers to research question four: *"Can semi-supervised and unsupervised strategies be devised on specialist, fine-grained datasets such as grass density and composition estimation?*. We specifically proposed a low supervision baseline for dry grass clover biomass prediction that makes use of unlabeled images. To do so, we first trained a herbage height aware semantic segmentation network on synthetic images that we then used to generate automatic labels for the unlabeled data using a small set of labeled images. We then trained a regression CNN on RGB images directly using the automatic labels to improve the accuracy over using the trusted data alone. This means that although expensive semantic segmentation has to be performed to generate labels for the unlabeled images, it is not required at inference time. We demonstrated the importance of our herbage height aware segmentation network when predicting dry herbage masses from canopy view images as well as the noise robust mechanisms we use to train on automatically labeled data. We improved over our baseline on the Irish dry herbage biomass dataset and set a new state-of-the-art performance level on the publicly available GrassClover dataset.

In practice, the algorithm could be deployed on phones with farmers taking pictures of the fields as they walk their farm. Deploying the grass composition prediction model on phone devices would provide a non destructive tool to quickly estimate the clover composition of grass and take appropriate fertilization steps if necessary. We found in this chapter that our proposed grass composition prediction model manages to generate to phone images at the cost of a slight increase in prediction error. Future solutions should look into using the available unlabeled phone data to improve the generalization capabilities.

For future work, unmanned image capture means such as drones could be used to save farmers time by limiting the amount of time spent walking the farm. Although drone images offer a definite advantage because of the large areas covered in each image, large areas also mean that evaluating ground-truth will become increasingly more complicated as large grass areas will have to be cut down and manually separated to acquire the biomass information. This paradigm means that low supervision approaches will become mandatory to move to image acquisition using drones. Chapter 7 will explore this possibility.

Chapter 7

Semi-supervised domain adaptation from camera to drone images for dry herbage biomass estimation

This chapter presents a semi-supervised approach to predict herbage characteristics from drone images. The chapter will detail how an unpaired GAN was used to substantially improve the resolution and quality of drone images. The improved drone images are then used together with labeled high-resolution images captured on the ground to train a semi-supervised algorithm that generalizes to drone images without the need for additional annotations. Section 7.1 motivates the benefits and challenges of using drones to predict herbage characteristics. Section 7.2 presents a detailed description of proposed super-resolution and semi-supervised regression approaches. Section 7.3 contains an ablation study, comparisons against the state-of-the-art and the work proposed in Chapter 6. Section 7.4 concludes the research carried out in the chapter with a discussion on future work. The research that emanated from this work was published at the AgricultureVision workshop at the 2022 IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR).

7.1 Motivations

Knowing the herbage composition and dry mass is valuable for the farmer but because existing probing processes are destructive and time consuming it is never evaluated at the farm level. In this context, deep learning has the capacity to provide a simpler, non-destructive alternative to dry herbage phenotyping and mass estimation from images alone. The feasibility of the method has been shown in Chapter 6 where the algorithm presented proposed to apply deep learning algorithms to ground-level images using handheld devices and tripods [75] or all terrain vehicles (ATV) [175]. In this chapter, we propose to extend the dry biomass and herbage mass estimation problem to drone images, which are more suitable for covering large herbage fields. Because drones operate at higher altitudes, large land areas that can span from tens to hundreds of square meters depending on the altitude are captured in every drone image, rendering the fine ground-truthing of data very challenging. To mitigate this issue, we propose to transfer knowledge learned from few high-resolution ground-level images to drone images in an unsupervised manner. To do so, we apply an unpaired domain transfer algorithm [148] to the drone images to enhance their resolution to 2048×2048 and to reduce the visual domain gap with the ground-level images (see Figure 7.1). We then train a semi-supervised neural network for regression on a small number of labeled ground-level images together with unlabeled

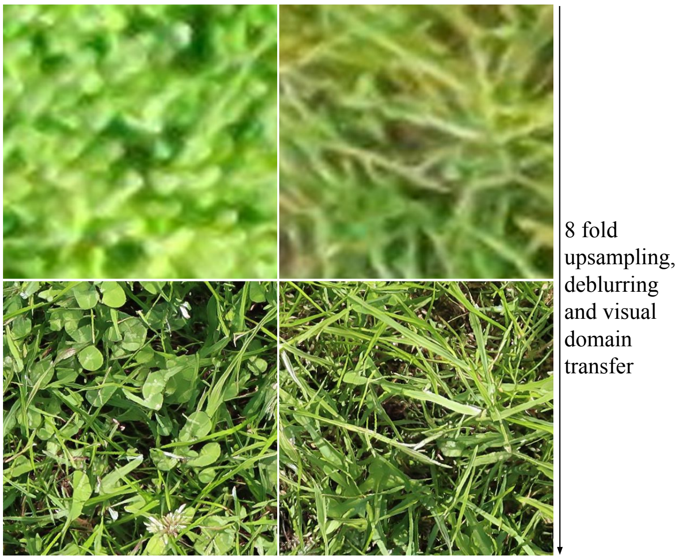

Figure 7.1: Up-sampling drone images by a factor of 8. Images at the top are 64×64 crops from drone images. Images at the bottom are up-sampled to 512×512, deblurred and transferred to the ground-level visual domain in an unpaired fashion. We use the transformed images in a semi-supervised regression objective.

drone images to effectively transfer knowledge between the two domains. To evaluate the quality of our regression algorithm, we test it on a small data set of ground-truthed drone images collected in Ireland and evaluate the benefit the large quantities of unlabeled images drone imagery provides to improve the ground-level predictions.

7.2 Unsupervised domain adaptation and super resolution on drone images

We aim to solve the biomass prediction task jointly from a small set of ground-level images $\mathcal{X}_l$ with biomass labels $\mathcal{Y}_l$ (ground-level images) together with a large set of unlabeled (raw) images $\mathcal{X}_u$ from a different visual domain (drone images) in an unsupervised fashion. To do so, we use two neural networks: Ω

performing super resolution and visual shift from the domain of $\mathcal{X}_u$ to $\mathcal{X}_l$ and Φ, a regression network we use to learn jointly from $\mathcal{X}_l$ and $\mathcal{X}_u$ by optimizing a semi-supervised objective. Contrary to the semi-supervised algorithm of Chapter 6 the goal here is to devise a simpler approach were the automatic labels are guessed using the same network as the one used to estimate the biomass composition and mass Φ.

7.2.1 Drone images for the Irish dataset

We propose in this chapter an extension of the Irish dataset presented in section 6.2.2. We collect drone images in the same 23 herbage paddocks originally studied in Ireland in late Autumn of 2021. We collect between 36 and 7 drone images per paddock at an altitude between 6 and 12 meters. The drone we use is the DJI Mavic 2 Pro [1] with its default camera, taking pictures at a resolution of 5472×3648. Although our drone is not capable of capturing its altitude relative to the land below, we subtract the above sea level GPS altitude of the drone from the land altitude at the associated GPS coordinates to obtain an approximate relative altitude using an open source API [2]. We obtain 328 drone images in total with their associated altitude. Because of the huge areas covered by drone images, the ground-truth we collect is limited to the dry herbage mass at the paddock level and we omit the grass height and biomass percentage information. The resulting 80 labeled drone images are only used as a means to test the knowledge transfer from the ground-level to the drone images and not used for training. We propose two ground-truth estimations for the drone images: the first is a visual estimation performed on

[1] https://www.dji.com/ie/mavic-2

[2] opentopodata.org

site at the time of the image collection by two human experts, very familiar with the site and that visually estimate the herbage on site every week. The second is obtained following the protocol of Egan et al. [53]: we cut two 1.2×8 meters strips in the paddocks 4 cm above ground level (typical cow grazing height) using an Etesia lawn mower (Etesia UK. Ltd., Warwick, UK). A 100 grams sample is collected from the cut material and dried at 95°C for 16 hours to obtain the dry herbage mass. We compare our algorithm against the human estimation and the exact ground-truth.

7.2.2 Contrastive Unpaired Translation (CUT)

The first step of our algorithm is to increase the resolution of drone images and to modify them to appear visually closer to the few ground-truthed images captured using high resolution cameras on the ground. To do so, we use Contrastive Unpaired Translation (CUT) [148] that we train from scratch on ground-level and drone grass images. CUT trains an adversarial network (GAN) to perform unpaired image style transfer using three principal components. G is the generator part of the network, competing to fool D the discriminator in an alternative adversarial optimization and F the projection head is used to optimize the contrastive part of the algorithm, which promotes semantic similarities between the same image before and after the visual transformation. CUT minimizes a combination of three losses to learn the parameters for G, D, and F. First the adversarial loss [64]

$$\begin{aligned}\mathbf{L}_{adv}(G, D, \mathcal{X}_l, \mathcal{X}_u) &= \mathbb{E}_{x_l \sim \mathcal{X}_l} \log D(x_l) \\ &\quad + \mathbb{E}_{x_u \sim \mathcal{X}_u}(1 - \log D(G(x_u))),\end{aligned} \tag{7.1}$$

promotes the generator G to transform images from $\mathcal{X}_u$ (the drone images) so that they become indistinguishable by the discriminator D from the high resolution ground-level images in $\mathcal{X}_l$. Second, once the image has been transformed by the generator, a patch contrastive regularization objective is applied where patches at the same location in the image before and after the transformation are encouraged to have similar features after projection through F while being dissimilar to any other random patch from the image. This results in a constrastive patch objective

$$\mathbf{L}_{patch}(G, F, X) = -\frac{1}{P}\sum_{i=1}^{P} \log\left(\frac{\exp\left(ip(p_i, p_i')/\tau\right)}{\sum_{k=1}^{P}\exp\left(ip(p_k, p_i')/\tau\right)}\right), \tag{7.2}$$

where $P = 64$ random patches are cropped out from the input image, their feature representations encoded through G (stopping half way), projected through F and $L2$ normalized. The process is repeated for the transformed version of the image to form P pairs of random patches $\{(p_i, p_i')\}_{i=1}^{P}$ for a given image $x \in \mathcal{X}_u$ where p_i is the representation before the domain shift and p_i' after. The dot product between the representations of corresponding pairs is encouraged to be close to one and close to zero for different patches. $\mathbf{L}_{patch}$ can also be applied to images in $\mathcal{X}_l$ to enforce that G will perform the identity operation, *i.e.* $\forall x \in \mathcal{X}_l, G(x) = x$. $\tau = 0.07$ is the contrastive temperature parameter. The final objective minimized by CUT where $\lambda_1 = \lambda_2 = 0.5$ is

$$\begin{aligned}\mathbf{L} &= \mathbf{L}_{adv}(G, D, \mathcal{X}_l, \mathcal{X}_u) \\ &\quad + \lambda_1\mathbf{L}_{patch}(G, F, \mathcal{X}_u) + \lambda_2\mathbf{L}_{patch}(G, F, \mathcal{X}_l).\end{aligned} \tag{7.3}$$

7.2.3 CUT for super resolution and style transfer

Cropping the drone images. We propose to crop squared areas from the drone images to obtain similar amounts of elements per image as ground level images. Because the drone images were not all captured at the same altitude, we adjust the area cropped out from the drone images depending on the altitude at which the image was captured. We observe visually that at an altitude of 8 meters, a 256×256 pixel crop of the drone data yields similar numbers of grass elements and of similar size to the ground-level images. Given the altitude of the drone at the time the picture was taken, we multiply the edge of the crop by the ratio between the altitude and the standard value of 6 meters, *i.e.* for an altitude of 12 meters, the edge of the square crop will be $6/12 \times 256 = 128$. Figure 7.2 illustrates the height adjusted cropping process. This process allows us to capture the same area of land independently of the height of the drone.

Deblurring the crops Although CUT is originally designed to transfer styles between two unpaired visual domains, we propose here to task the algorithm with improving the resolution of the drone images while at the same time transferring their visual style to ground-level images. Note that the super-resolution task is usually performed by conditional GANs (e.g. [139]) but we propose here to use an unpaired algorithm. For each image $x \in \mathcal{X}_u$, we upscale the image from the original resolution to 2048×2048. Ω is then trained to transfer the visual style of the ground-level high resolution images to the up-sampled crops, effectively deblurring them to appear closer to the higher resolution images (see Figure 7.1).

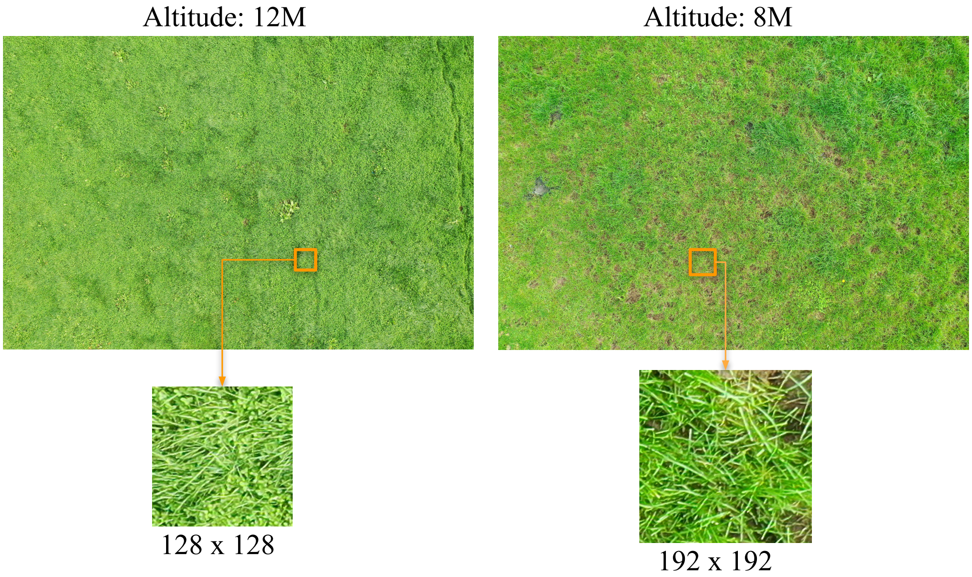

Figure 7.2: Drone image cropping process at different altitudes. Given that resolution of the image is fixed, we increase or reduce the cropped area. All crops are then bicubicly upscaled to 2048×2048 before deblurring.

7.2.4 Semi-supervised regression on drone data

By up-sampling and visually transforming drone images to appear closer to the ground-level visual domain, we are now able to learn jointly from $\mathcal{X}_l$ and $\mathcal{X}_u$. Since it is only practical to obtain labels for ground-level camera images, we propose to optimize a semi-supervised regression objective using $\mathcal{X}_l$ as the labeled set and $\mathcal{X}_u$ as the unlabeled data. After an initial pretraining of Φ on $\mathcal{X}_l$, we start guessing biomass labels for X_u using a consistency regularization approach [18]. Using two data augmented views x'_u and x''_u (vertical and horizontal random flipping), we use an exponential moving average (EMA) on the weights of Φ to guess two approximate biomass labels y'_u and y''_u for x_u. Rather than averaging the two approximate labels with equal importance like consistency regularization algorithms for image classification [18, 19], we draw a random mixing parameter λ from a uniform distribution to improve the regularization of the predictions and avoid confirmation bias [9]. We obtain

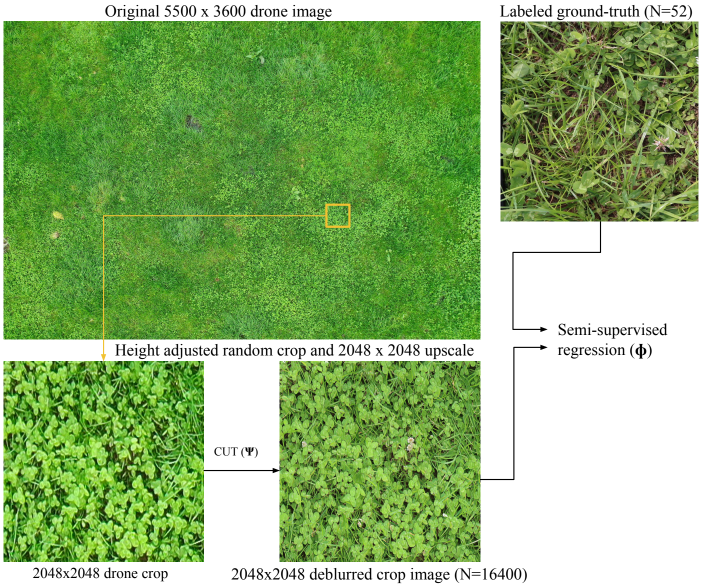
Original 5500 x 3600 drone image
Labeled ground-truth (N=52)
Semi-supervised
regression (ϕ)
Height adjusted random crop and 2048 x 2048 upscale
CUT (Ψ)
2048x2048 drone crop
2048x2048 deblurred crop image (N=16400)

Figure 7.3: Overview of our up-sampling and knowledge transfer algorithm. We use an up-sampling and visual domain transfer network Ω and a semi-supervised network Φ that we use to learn jointly from few labeled ground-level examples ($N = 54$) and unlabeled drone images.

an approximate label $\hat{y} = \lambda y'_u + (1 - \lambda)y''_u$ for every unlabeled images. We enforce the distribution of the predictions on unlabeled samples to match the observed ground-truth distribution on the labeled data (distribution alignment) by multiplying the label prediction by the ratio between a sliding window average (50 mini-batches in practice) of $\hat{y}$ and the observed distribution on the labeled data. EMAs and distribution alignment are common principles of consistency regularization algorithms for semi-supervised learning [88, 19]. Finally, we normalize the biomass composition to sum to 1 in the approximate label $\hat{y}$. Figure 7.3 presents an overview of the proposed semi-supervised training algorithm. Section 7.3.5 will compare the accuracy of this semi-supervised strategy against the algorithm proposed in Chapter 6.

7.2.5 Regression from images

We predict biomass labels from images in $\mathcal{X}_l$ and $\mathcal{X}_u$ using Φ to extract visual features. For the Irish dataset [75], we use three different linear heads, separating the predictions of the herbage mass, herbage height, and biomass composition. We normalize the herbage mass and herbage height values between 0 and 1 using fixed normalization values (4000 kg DM/ha for the herbage mass and 20cm for the height) and offset them by $+0.2$ to improve the prediction for low values originally too close to 0. To obtain values between zero and one for each target prediction and ensure that the sum of the biomass percentages equals one, we apply a softmax function on the three outputs from the biomass head and a sigmoid function for each of the other two values. For the GrassClover dataset [174], we use a single linear head, predicting the biomass percentages (grass, white clover, red clover, weeds) and sum the predictions for white and red clover to obtain the total clover content. These configurations follow the work of Albert et al. [3]. We use the root mean squared error (RMSE) as the training objective.

7.3 Experiments

7.3.1 Experimental setup

We conduct experiments on two biomass prediction datasets from canopy images. For the GrassClover dataset, we use 100 labeled and $1,000$ unlabeled images for training and 57 images for validation. We report the test accuracy results on the evaluation server for the GrassClover dataset. For the Irish dataset, we use 52 labeled and 595 unlabeled images for training and 104

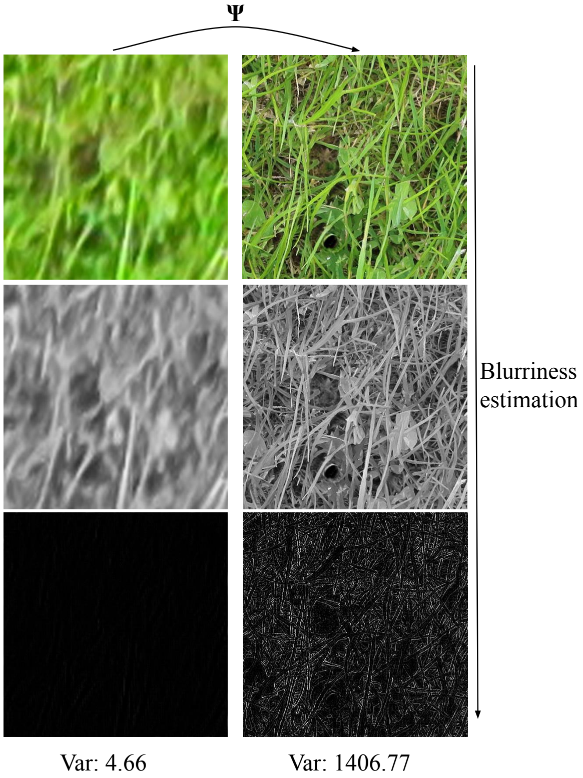

Figure 7.4: Overview of the deblurring effect of Ω on drone data. A high variance indicates a sharper image.

images for validation. For the drone images, we extract 50 random crops from each of the 328 images to create a dataset of $16,400$ unlabeled images. We train using stochastic gradient descend at a resolution of 512×512 with a batch size of 32 and a fixed learning rate of 0.03. We update the EMA with a multiplication parameter of 0.99 at every mini-batch. The training augmentations are resize, random crop, random horizontal and vertical flipping, and normalization. When we perform semi-supervised learning, we create each mini-batch by aggregating 4 labeled samples with unlabeled images as in Chapter 6. For the neural networks, we use a ResNet18 [92] pretrained on ImageNet [100] for the regression network Φ and the 9 ResNet blocks version of the CUT model Ω.

7.3.2 Drone image deblurring and style transfer

We evaluate the deblurring capacity of Ω by computing the variance of the Laplacian on the grayscale view of an image. Computing the Laplacian of the image allows us to extract edges in the image and the variance of the resulting value quantifies the sharpness of the edges: $\text{sharpness} = \mathbf{var}(\nabla^2(\text{grayscale}(\text{image})))$ [152] with ∇^2 the Laplacian operator. A higher variance indicates a better defined (sharper) image. The sharpness estimation process is illustrated in Figure 7.4. We observe that the variance averaged over all the crops changes from 5.32 when cropping directly from the drone images to 1261.05 for the same images deblurred by Ω.

7.3.3 Semi-supervised biomass, herbage height and herbage mass prediction

We evaluate the capacity of our algorithm to predict the biomass composition of herbage (%) together with an estimation of the dry herbage mass (kg DM/ha) and the grass height (cm) in a semi-supervised manner. We run experiments on the Irish dataset with the original unlabeled images but also study replacing them with an equal number ($N = 596$) of deblurred drone images. This is to evaluate the capacity for drones to capture unlabeled data that can be used to improve the prediction at the ground-level. For evaluation purposes, we compute the Herbage Root Mean Square Error (HRMSE) which is the RMSE between predicted and ground-truth herbage mass for grass, clover, weeds and for the total mass and the Herbage Relative Error (HRE) which is the ratio between the ground-truth value of the total herbage mass and the prediction: $HRE = \frac{pred_{herbage}}{gt_{herbage}}$. The HRE measure is typically used to compare human visual estimation against the collected

Table 7.1: Ablation study and comparison against state-of-the-art algorithms on the Irish dataset. The last row denotes replacing the ground-level unlabeled camera images with deblurred drone images. The best results are in bold.

	HRMSE						RMSE				
	Total	Grass	Clover	Weeds	Avg.	HRE	Grass	Clover	Weeds	Avg.	HE
Chapter 6	230.10	220.84	34.86	27.13	94.28	1.14	4.81	4.75	3.42	4.33	2.15
Albert et al. [5]	229.12	218.02	37.65	29.21	94.96	1.09	**4.58**	4.22	3.44	**4.08**	**2.03**
Labeled only	229.23	268.90	107.39	39.82	138.71	1.08	17.15	14.08	4.74	11.99	2.28
Semi-sup	234.50	224.10	43.03	26.74	97.96	1.04	5.85	5.51	**3.19**	4.85	2.24
+ distribution alignment	220.79	215.88	40.00	26.78	94.22	1.08	5.53	5.51	3.25	4.76	2.09
+ EMA	217.28	208.96	34.16	**26.50**	89.88	1.08	4.86	4.73	3.26	4.28	2.09
+ Unsup init [107]	211.57	202.18	**28.93**	26.80	**85.97**	1.09	4.54	4.50	3.21	**4.08**	2.09
drone unlabeled	**209.69**	**199.61**	33.59	27.13	86.78	**1.02**	4.74	4.65	3.33	4.24	2.18

ground-truth [53]. We additionally compute the RMSE over the predicted percentages of grass, clover, weeds in the herbage and the Height Error (HE) which is the RMSE between the predicted herbage height and ground-truth height. We compare against state-of-the-art results on the Irish dataset where the validation images are ground-level images in Table 7.1. We study the importance of the different elements of the semi-supervised algorithm on the validation error, including enforcing distribution alignment for the label guesses and using an exponential moving average (EMA) on the weights of the semi-supervised network. We also report results when initializing the weights of the network using an unsupervised representation learning algorithm [107] on the unlabeled data as in Albert et al. [5].

We finally point out that using an equal number of deblurred drone images produces comparable results to using the original unlabeled images (last two rows in Table 7.1). This result motivates the use of drone images to easily capture large amounts of unlabeled images. Figure 7.5 shows a line plot of the HRE compared against the ground-truth herbage mass where we observe

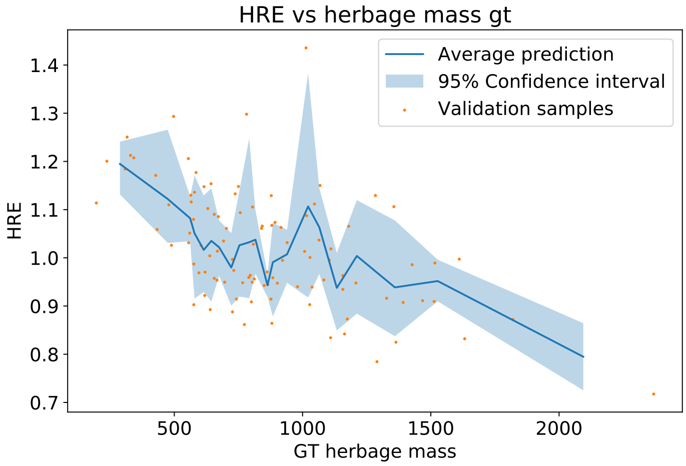

Figure 7.5: Visualization of the HRE on the validation set of the Irish dataset with 95% confidence intervals.

	HRMSE	HRE
Against harvested ground-truth		
Labeled only	1094.18	0.43
Semi-sup.	566.68	0.81
Semi-sup. drone	219.15	0.97
Human expert	170.03	1.04

Table 7.2: Results on drone images. Errors are computed against the absolute harvested ground-truth.

that the algorithm struggles the most on high or low herbage mass outliers (< 500 to > 2000 kg DM/ha). This is most likely due to the low amount of high or low herbage mass examples seen during training.

7.3.4 Prediction on drone images

Table 7.2 reports the Herbage Root Mean Square Error (HRMSE) when predicting on drone data without the need to gather additional labels. First, we evaluate the accuracy of a CNN model learned on ground-level data only to predict on deblurred drone patches using Ω. We then evaluate the accuracy benefits of training on deblurred drone patches in a semi-supervised manner

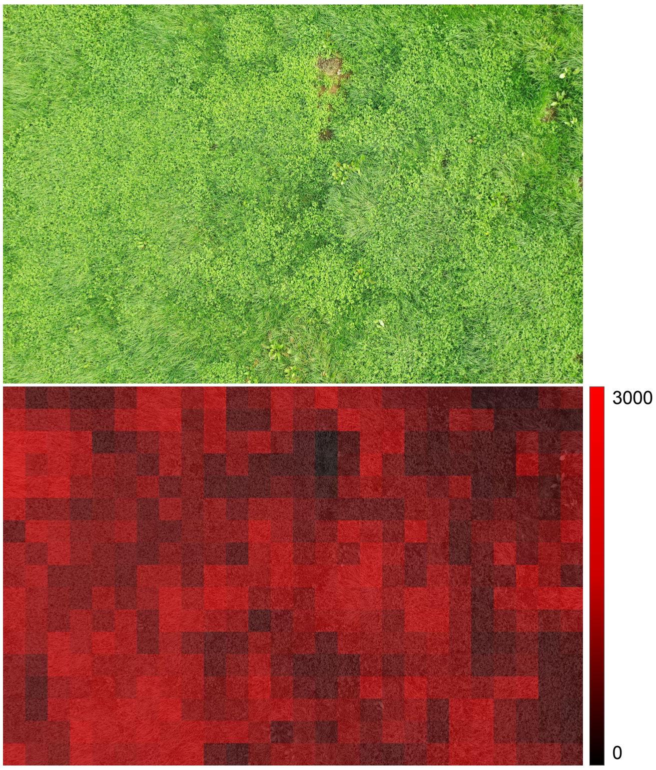

Figure 7.6: Prediction on ordered crops of the drone images. Paddock level herbage mass ground-truth: 1735 kg DM/ha, average prediction: 1719.21 kg DM/ha. Altitude 8.2 meters. The ground-level area covered by each crop is approximately $.5 \times .5$ square meters. Note the dirt patch with no grass at the top of the image where a very low expected dry herbage mass is predicted by our algorithm.

for the herbage mass prediction and compare the error rate of our algorithm against human experts. We observe a significant reduction in error rates when the drone images are used in the semi-supervised objective (semi-sup. drone row) over using only the ground-level data (labeled only or semi-sup. with the unlabeled ground-level images). The performance proposed by our low supervision algorithm is close to be on par with human experts at the paddock level. Figure 7.6 illustrates the prediction process at the image crop level for a given drone image where we predict the expected dry herbage mass per ha. We also report the prediction histogram for each drone image crop

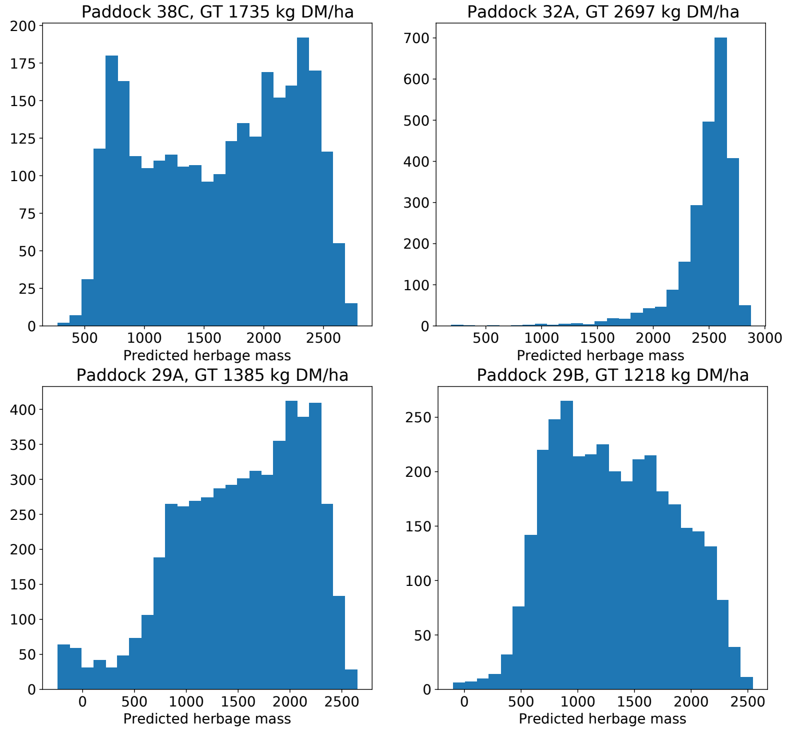

Figure 7.7: Histograms of the prediction on drone image crops of a paddock.

of a paddock when predicting the herbage mass in Figure 7.7. The final prediction is obtained by averaging the prediction over all enhanced crops of drone images of a given paddock. We observe in Figure 7.7 that the grass density can vary a lot across the same paddock and that multiple pictures are necessary to cover the full paddock and to get an accurate estimation.

Table 7.3 reports on how augmenting the number of random crops improves the prediction of the best performing model. We report the average herbage mass error and standard deviation over 5 random sets of crops. “All” denotes cropping the image in a checkerboard fashion and using all crops (from 250 to $1,000$ crops per image depending on the altitude). Although 1 random crop per image yield interestingly good results, we validate our choice of 50 crops per images for more stable predictions.

Table 7.3: RMSE errors on drone image for varying amounts of random crops per image for the best performing model. Averaged over 5 random sets of crops.

	1	5	20	50	100	all
HRMSE	252.71 ± 51.24	227.18 ± 20.15	222.97 ± 12.56	219.15 ± 6.93	220.09 ± 4.14	219.53

		Clover				
	Grass	Total	White	Red	Weeds	Avg.
Skovsen et al. [174]	9.05	9.91	9.51	6.68	6.50	8.33
Naranayan et al. [135]	8.64	8.73	8.16	10.11	6.95	8.52
Chapter 6	8.78	8.35	**7.72**	**7.35**	7.17	7.87
Labeled only	9.81	8.49	7.99	8.58	7.25	8.57
Semi-sup.	**6.68**	**7.76**	8.08	8.66	**6.72**	**7.58**

Table 7.4: Results on the GrassClover test set (RMSE). Lowest errors are in bold.

7.3.5 Semi-supervised biomass prediction on the GrassClover dataset

We compare our semi-supervised approach against state-of-the-art algorithms on the publicly available GrassClover dataset in Table 7.4 where we report RMSE errors for the biomass percentage prediction on the held out test set[3] where we perform on par with existing approaches.

7.3.6 Comparison with Chapter 6

Both algorithms in Chapter 6 and in this chapter use unsupervised images to improve the accuracy of grass composition estimation algorithms. The algorithm in this chapter is inspired from semi-supervised image classification algorithms used in Chapter 3 but where the image augmentations have to be

[3] https://competitions.codalab.org/competitions/21122

adapted to avoid corrupting the ground-truth in the case of image regression. The advantage of the semi-supervised algorithm in this chapter is that it does not require any additional model to guess approximate labels for the unlabeled images (segmentation model Ψ in Chapter 6). The approximate label guesses for the unlabeled images are also refined every epoch by the regression network Φ, which explains the small improvements over Chapter 6 in Tables 7.1 and 7.4.

7.4 Conclusion

This chapter aims to provide answers to research question five: *"Can super-resolution and semi-supervised learning be applied to generalise a grass composition prediction model learned on ground-level images to drone data?"* We investigated how to extend the biomass estimation and herbage mass prediction problem from ground-level studied in Chapter 6 to drone images. By its nature, the herbage biomass information of drone images is hard to annotate finely because of the huge areas covered. Ground-level data, however, has the advantage of providing easier to acquire, finely annotated, and high resolution images of the herbage but would be a limited solution to generalize targeted fertilization on entire herbage fields. To successfully transfer knowledge from the ground-truth images captured on the ground to the drone data, we proposed to train an unpaired style transfer algorithm to deblur height adjusted crops of drone images. To do so, resolution is increased by a factor of 8 and the visual style of ground-level images captured using different cameras is transferred to the drone images. The large set of enhanced but unlabeled drone images is used together with the finely annotated ground-level images to learn unsupervised initialization weights and to train a semi-supervised regression algorithm. The neural network trained on the partially labeled set largely improved the regression accuracy on the ground-level data and the herbage mass prediction on drone images. When evaluating the trained neural network on a small set of ground-truthed images at the paddock level, we significantly reduced the prediction gap with human experts, achieving error rates close to experienced technicians familiar with the land. This early results are encouraging for future research in the field of low supervision computer vision for herbage biomass prediction from drone images. The semi-supervised algorithm we propose can also be applied to the publicly available GrassClover dataset where we further decrease the biomass composition error when compared to the algorithm of Chapter 6, consolidating the previous answer to research question four on the feasibility of low supervision solutions to the grass composition estimation problem.

Currently, the main limitation of the research is the limited amounts of data the algorithm is tested on. Further data collection should be carried out, not so much for increasing the amount of training samples but to add variety to the grass image test set. Collecting and ground-truthing drone images over multiple physical locations in Ireland and across different seasons/years seems mandatory to validate the findings of this chapter. Another limitation is the sensitivity of the algorithm to the dynamic crop adjustments. In the scope of this research, the height of the drone at the time of the image capture was estimated using GPS coordinates yet an embarked sensor would be a more accurate solution. Finally concerning pure deep learning algorithm design, the approximate predictions for the dry herbage made by the human experts could be used as an approximate but easy to acquire ground-truth that could be used to improve results. To avoid analysing the full drone image, this chapter studied making a prediction using only a random subset of all crops of an image and found that even with a limited amount of crops, accurate predictions can be achieved and effectively decrease the computational needs of the algorithm. In future research, a smart algorithm could be designed to sample areas of interest in the drone images to reduce the variability due to the random sampling.

Chapter 8

Conclusion and future work

8.1 Answers to the research questions

RQ1: What is the nature of web noise and can detected noisy images be included in the training objective?

In chapter 3 and 4, strategies have been used to detect and utilize out-of-distribution noise in web datasets. In both cases, using out-of-distribution images in the training objective has been beneficial for the classification accuracy on the held-out test set. In chapter 2, the DSOS algorithm assigns a uniform label to out-of-distribution images. This strategy was shown to reduce the calibration error of the neural network as well as promoting high entropy prediction on seen and unseen out-of-distribution data, enhancing the rejection of these noisy images by the neural network. In chapter 3, the SNCF algorithm minimizes a contrastive loss term on the out-of-distribution images. Similar out-of-distribution images are clustered together while pure outliers are trained on in an unsupervised manner. Including the out-of-distribution data in the contrastive objective is shown empirically to improve the classification accuracy over ignoring the noisy samples. Both chapters demonstrate that on datasets of a relatively small size (<1M images), strategies can be devised to improve the classification accuracy using out-of-distribution data

RQ2: Can unsupervised learning be used to detect noise in web-crawled datasets?

Experiments conducted in chapter 4 have shown that unsupervised contrastive learning can separate out-of-distribution images from in-distribution ones on the hypersphere projection used. In the case where no human supervision is available to separate the out-of-distribution from in-distribution images, clustering can be performed on unsupervised features transformed using a spectral projection. The separation observed on artificially corrupted datasets does not generalize as well to web data, where further research should be conducted.

RQ3: Can unsupervised features be used as a medium to propagate labels in a semi-supervised scenario when few labels are available?

Chapter 5 studied the propagation of labels from labeled to unlabeled images using unsupervised features. Although using all propagated labels directly was shown to be a limited solution, selecting a trusted subset of the propagated labels using a label noise detection approach experimentally proved to be a suitable manner to augment the size of the labeled subset. Once the size of the labeled subset is augmented, state-of-the-art semi-supervised learning algorithms can be applied and will reach higher classification accuracies because more labeled examples are available.

RQ4: Can semi-supervised and unsupervised strategies be devised on

specialist, fine-grained datasets such as grass density and composition estimation?

Chapter 6 and 7 applied semi-supervised and unsupervised algorithms to reduce the data collection effort needed to train a regression model to predict herbage composition. This is a real world problem where image ground-truthing is time consuming and destructive. In the case of chapter 7, semi-supervised learning was used to reduce the annotation effort 8 folds with little degradation in composition prediction error where unlabeled images were successfully used to divide the prediction error by three (8 absolute points) on an Irish grass composition dataset and by 1 absolute point on a publicly available Danish dataset proposing a challenging fine-grained detection between red and white clover detection. Chapter 6 proposed a semi-supervised strategy more specific to the grass composition estimation from canopy images problem but observed an equally significant reduction in prediction error when using unlabeled images. These two chapters studied training deep learning models on specialist datasets where ground-truth is difficult to collect and with real world applications. In both cases, low supervision algorithms inspired by the state-of-the-art on curated datasets successfully used unlabeled images to reduce the regression error on a held out test set.

RQ5: Can super-resolution and semi-supervised learning be applied to generalise a grass composition prediction model learned on ground-level images to drone data?

Chapter 7 proposed to use an unpaired generative network to learn to enhance images from blurry and saturated drone images to high resolution corrected images that were visually comparable to images captured using a high resolution camera close to the ground. Although a regression model trained on the high resolution camera images only was not able to generalize directly to the enhanced drone data, a semi-supervised training was proposed to include unlabeled drone images in the training stage. Even if no ground-truth was used for the drone images, training in a semi-supervised way using ground-truth for camera images only enabled reaching satisfactory accuracies on a held-out drone image test set.

8.2 Research contributions

The per-chapter research contributions are as follows:

Chapter 3

1. A representative survey over the type of noise to be expected when constructing a dataset using web queries.

2. A novel noise detection metric, entropy of the interpolation of the network prediction and the ground-truth label, that is capable to accurately differentiate between clean, ID and OOD noise.

3. DSOS, a simple algorithmic solution to combat ID and OOD noise in web-crawled datasets validated using controlled experiments and ablation studies on corrupted versions of the CIFAR-100 dataset.

4. A comparison of DSOS against state-of-the-art, noise-robust algorithms on real-world web-crawled datasets, demonstrating the validity of our findings for real-world applications.

Chapter 4

- A dual noise detection approach utilizing the alignment and uniformity principles of contrastive learning to detect noisy samples using a spectral embedding of unsupervised representations.
- A noise robust algorithm capable of training a CNN on a dataset corrupted with in-distribution and out-of-distribution noise, correcting the label of in-distribution whilst using out-of-distribution noise to improve low-level features.
- Experiments on controlled and real world noisy datasets demonstrating the state-of-the-art performance of our algorithm.

Chapter 5

1. An unsupervised knowledge-bootstrapping pipeline that enhances the performance of semi-supervised algorithms when very few labeled samples are available.
2. A reliable sample selection method in the presence of label noise induced by label propagation. The method is robust to class and noise imbalance.
3. We evaluated the importance of good unsupervised features for label propagation, and demonstrated the superiority of our approach when dealing with feature-based label noise generated by label propagation.

Chapter 6

1. A herbage height aware, weakly supervised, semantic segmentation algorithm trained on synthetic images that is used to automatically label data.

Research published in the Agriculture-Vision Workshop at the Conference on Computer Vision and Pattern Recognition (CVPR) 2022

2. An algorithm leveraging automatically labeled images to improve grass/clover/weed dry herbage mass estimation.

3. A detailed study of the importance of the low supervision elements for the final accuracy of our algorithm, and a comparison with the state-of-the-art on a publicly available dataset.

Chapter 7

1. 328 drone images of herbage fields in Ireland.

2. An unpaired image transfer pipeline, increasing the resolution of drone images 8 fold and transferring them to the ground-level camera visual domain.

3. A semi-supervised regression that learns to estimate dry herbage biomass from a small set of annotated ground-level images and unlabeled drone images.

8.3 Future research areas

Here are listed the future areas of interest on the topic of low supervision for image classification research that I believe should be studied further.

- **Noisy vs hard sample for deep learning in label noise scenarios.** Chapter 3 detects noisy samples relying on how hard they are to fit for the network and Chapter 4 relies on the unique visual characteristics of out-of-distribution samples. Both of these specificities are also shared by hard (but clean) training samples. Being able to differentiate between noisy and hard samples is important to avoid rejecting training examples which could help generalize better, especially to edge cases.

- **Utilizing detected out-of-distribution samples in web-crawled datasets.** Chapters 3 and 4 showed that strategies could be devised to include out-of-distribution images during training to improve the validation accuracy of a neural network trained on web-crawled data. These improvements might be limited due to the small size of the datasets studied in this thesis, and in the case where large amounts of images are available in each class, removing at least some of the out-of-distribution noise might be preferable. Further research should evaluate where the size limit is and if some out-of-distribution samples are more relevant to keep in an unsupervised term than others to bridge knowledge gaps in under-represented classes.

- **Evaluation of low supervision algorithms on uncurated datasets.** A limitation of how new semi- or unsupervised algorithms are compared to each other is the highly curated datasets upon which they are compared. Even if image labels are artificially removed on datasets such as ImageNet or CIFAR, the unlabeled images will still be high visually relevant to the classification task. In real world applications, the unlabeled images would tend to be gathered in bulk from the web, for example, and some would be irrelevant to the classification task. STL-10 used to be a more realistic dataset in this sense, but its adoption has been reduced because the amount of labeled data is large compared to the current semi-supervised performance. Evaluating unsupervised algorithms on WebVision vs ImageNet would also be an interesting comparison to include in future research.

- **Further validation of the grass composition estimation algorithms.** A limitation of the work presented in chapters 6 and 7 is the limited amount of available validation samples. More data collection should be conducted over multiple seasons and geographical locations in Ireland. If the current algorithms fail to generalize, the domain gap should be bridged using unlabeled images to limit the data collection effort.

8.4 Conclusion

The research conducted during this PhD focused on low supervision alternatives for computer vision using deep learning. The first two chapters focused on label noise and datasets directly crawled from the web with no human curation.

Chapter 3 conducted a study on the WebVision dataset to identify the type and quantity of label noise to be expected in web crawled datasets. Out-of-distribution noise was identified as the dominant noise type with in-distribution noise being also present in limited amounts. In order to design an algorithm robust to both noise types, a novel metric was proposed to independently retrieve in- and out-of-distribution noisy samples. Once the noisy samples are identified, the true label of in-distribution samples is guessed in a semi-supervised way, while the network is encouraged to predict maximum entropy labels on out-of-distribution samples to promote rejection.

Chapter 4 also studied image classification on uncured web data, but proposed to use unsupervised learning to detect the noisy samples. Out-of-distribution samples were empirically observed to linearly separate from in-distribution ones on the hypersphere projection used in unsupervised contrastive learning algorithms. In order to keep noise retrieval unsupervised, spectral projection and OPTICS, a clustering algorithm, was used to retrieve the out-of-distribution cluster and in-distribution noise was identified as the outliers to the clean cluster. A guided contrastive algorithm was trained together with the classification objective so that similarities can be encouraged between in-distribution samples of the same class and between visually similar out-of-distribution images. Pure out-of-distribution samples (visually different from anything else in the dataset) are considered unlabeled (only similar to an augmented version of themselves).

Chapter 5 studied the combination of unsupervised learning and semi-supervised learning. The chapter focused on an alternative manner of using unsupervised learning other than network initialization or regularization. Unsupervised learning was used to learn similarities between labeled and unlabeled samples on a semi-supervised dataset. Image similarities were then used to propagate labels from the labeled to the unlabeled images using a diffusion algorithm. Training directly on all propagated labels was shown to be insufficient to improve the state-of-the-art classification accuracy, so another approach was devised where label noise detection was performed to select trusted samples to be added to the initially few labeled examples. By increasing the size of labeled pool of samples, state-of-the-art algorithms were empirically shown to benefit from a stronger supervised signal and converge to higher classification accuracies with no further human supervision required.

The last two chapters of this thesis investigated the application of low supervision techniques to real world agricultural problems. The proposed application is a regression task where the goal is to predict grass composition and dry weight from canopy images. One of the interesting aspects of this application is the collection of images in Ireland where I had a direct input into how the data gathering should happen in order to achieve an accurate

Research published in the Agriculture-Vision Workshop at the Conference on Computer Vision and Pattern Recognition (CVPR) 2022 deep learning model. To reduce the human effort required for data collection, a semi-supervised approach was motivated and large amounts of unlabeled images where collected.

In chapter 6, a first solution using unlabeled images was proposed. In order to compress the quantity of information present in canopy view images, a segmentation algorithm was used to segment pixels in the grass canopy and to predict the pixel level height of the grass. The pixel percentage for each species was then used to train a simple linear regression model to predict dry biomass percentages from species pixel percentages. To train the segmentation model with limited supervision, synthetic images were generated and used. Finally, pseudo-labels were predicted for the unlabeled grass images using the linear regression model and the final regression neural network was trained on the few labeled images together with the pseudo-labeled images using a label noise robust approach to avoid fitting incorrectly labeled images. The semi-supervised approach successfully reduces the prediction error by using the unlabeled images in a real world application.

Chapter 7 studied the same grass prediction problem as chapter 6 and further extended the prediction from ground-level images using cameras mounted on tripods to drones. In this case, a generic semi-supervised strategy for image classification was successfully adapted: consistency regularization. For the drone images, the few ground-truted images were kept for testing and only unlabeled drone images were available for training. A super-resolution and image deblurring approach was used to enhance the quality of the drone images so that they would look similar to ground-truth images captured on the ground. Consistency regularization was used to train in a semi-supervised manner on the ground-truth ground-level camera images and the visually enhanced yet unlabeled drone image crops. The validation done on the few labeled drone images demonstrated estimation accuracies on par with expert human estimation by eye.

In conclusion, designing low supervision alternatives to fully supervised approaches is highly desirable as it enables the distribution of deep learning solutions for computer vision to a wider audience by easing the constraints around dataset building. The research conducted so far has shown that in the case of image classification, unsupervised learning, semi-supervised learning, and label noise can be utilized in a synergistic fashion to maintain high classification accuracies when the image datasets are uncurated by humans, or when annotations are scarce. Semi-supervised algorithms proposed in the literature can be adapted, when proper care is taken, to the real world problem of grass composition estimation, and substantially reduce the amount of time needed to gather a specialist vision dataset.

For future work on label noise datasets, possible improvements to explore include: the investigation of different alternatives for the utilization of out-of-distribution samples dependent on the size of the dataset as out-of-samples might be useful to learn basic features on small datasets as shown in Chapter 4 but it might be better to ignore them in larger (1M+) image datasets. Further studies on the importance of hard examples when training on a web label-noise dataset as well as the separate detection of hard and noisy samples in web label-noise datasets could also be and interesting venue to study since most state-of-the-art label noise robust algorithms detect noisy samples because they are hard to fit for a neural network which is a similar learning parttern than clean but hard samples. For semi-supervised and unsupervised learning, more comparisons should be conducted on uncurated datasets so that a more realistic comparison is available. This could include a comparison when datasets are trained using few labeled samples from ImageNet together with a large unlabeled image pool from Webvision. Regarding the grass composition prediction problem, a limitation of the findings in this thesis remain the low quantity of test available data. The observed results should be validated further by collecting a large quantity of varied data over multiple seasons and possibly multiple locations. This would enable further testing of the unsupervised adaptability of the algorithms presented in chapter 6 and 7. In the case where generalization to the newly collected validation samples is insufficient, unsupervised images should be used to bridge the gap at a low data collection cost.

Bibliography

[1] Manya Afonso et al. “Tomato Fruit Detection and Counting in Greenhouses Using Deep Learning”. In: *Frontiers in plant science* (2020).

[2] Paul Albert et al. “ReLaB: Reliable Label Bootstrapping for Semi-Supervised Learning”. In: *International Joint Conference on Neural Networks (IJCNN)*. 2021.

[3] Paul Albert et al. “Semi-supervised dry herbage mass estimation using automatic data and synthetic images”. In: *IEEE International Conference on Computer Vision Workshops (ICCVW)*. 2021.

[4] Paul Albert et al. “Addressing out-of-distribution label noise in webly-labelled data”. In: *Winter Conference on Applications of Computer Vision (WACV)*. 2022.

[5] Paul Albert et al. “Using image analysis and machine learning to estimate sward clover content”. In: *European Grassland Federation Symposium*. 2022.

[6] Francisco Albornoz. “Crop responses to nitrogen overfertilization: A review”. In: *Scientia horticulturae* (2016).

[7] Mihael Ankerst et al. “OPTICS: Ordering points to identify the clustering structure”. In: *ACM Sigmod record* 28.2 (1999), pp. 49–60.

[8] E. Arazo et al. “Unsupervised Label Noise Modeling and Loss Correction”. In: *International Conference on Machine Learning (ICML)*. 2019.

[9] E. Arazo et al. “Pseudo-Labeling and Confirmation Bias in Deep Semi-Supervised Learning”. In: *International Joint Conference on Neural Networks (IJCNN)*. 2020.

[10] Devansh Arpit et al. “A closer look at memorization in deep networks”. In: *International Conference on Machine Learning (ICML)*. 2017.

[11] Y. M. Asano, C. Rupprecht, and A. Vedaldi. “Self-labelling via simultaneous clustering and representation learning”. In: *International Conference on Learning Representations (ICLR)*. 2020.

[12] Tewodros W Ayalew, Jordan R Ubbens, and Ian Stavness. “Unsupervised Domain Adaptation for Plant Organ Counting”. In: *European Conference on Computer Vision*. 2020.

[13] A. Babenko and V. S. Lempitsky. “Aggregating Deep Convolutional Features for Image Retrieval”. In: *European Conference on Computer Vision (ECCV)*. 2015.

[14] Adrien Bardes, Jean Ponce, and Yann LeCun. “Vicreg: Variance-invariance-covariance regularization for self-supervised learning”. In: *International Conference on Learning Representations (ICLR)*. 2022.

[15] Ruud Barth, Jochen Hemming, and Eldert J Van Henten. “Optimising realism of synthetic images using cycle generative adversarial networks for improved part segmentation”. In: *Computers and Electronics in Agriculture* (2020).

[16] Y. Bengio, O. Delalleau, and N. Le Roux. *Label propagation and quadratic criterion*. Tech. rep. Carnegie Mellon University, 2006.

[17] D. Berthelot et al. “MixMatch: A Holistic Approach to Semi-Supervised Learning”. In: *Advances in Neural Information Processing Systems (NeuRIPS)*. 2019.

[18] D. Berthelot et al. “MixMatch: A Holistic Approach to Semi-Supervised Learning”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2019.

[19] D. Berthelot et al. “ReMixMatch: Semi-Supervised Learning with Distribution Matching and Augmentation Anchoring”. In: *International Conference on Learning Representations (ICLR)*. 2020.

[20] Matteo Biasetton et al. “Unsupervised domain adaptation for semantic segmentation of urban scenes”. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2019.

[21] Avrim Blum and Tom Mitchell. “Combining labeled and unlabeled data with co-training”. In: *Annual Conference on Computational Learning Theory (COLT)*. 1998.

[22] Kushtrim Bresilla et al. “Single-shot convolution neural networks for real-time fruit detection within the tree”. In: *Frontiers in plant science* (2019).

[23] Nicolas Brunel, Vincent Hakim, and Magnus JE Richardson. “Single neuron dynamics and computation”. In: *Current opinion in neurobiology* (2014).

[24] Danilo Bzdok et al. “Semi-supervised factored logistic regression for high-dimensional neuroimaging data”. In: *Advances in neural information processing systems (NeurIPS)* (2015).

[25] M. Caron et al. “Deep clustering for unsupervised learning of visual features”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

[26] Mathilde Caron et al. “Unsupervised learning of visual features by contrasting cluster assignments”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2020.

[27] Mathilde Caron et al. “Emerging properties in self-supervised vision transformers”. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[28] L.-C. Chen et al. “Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation”. In: *European Conference on Computer Vision (ECCV)*. 2018.

[29] Minghao Chen, Hongyang Xue, and Deng Cai. “Domain adaptation for semantic segmentation with maximum squares loss”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2019.

[30] T. Chen et al. “A Simple Framework for Contrastive Learning of Visual Representations”. In: *International Conference on Machine Learning (ICML)*. 2020.

[31] Ting Chen et al. “Big self-supervised models are strong semi-supervised learners”. In: *arXiv: 2006.10029* (2020).

[32] Xinlei Chen and Kaiming He. “Exploring simple siamese representation learning”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021.

[33] Xinlei Chen, Saining Xie, and Kaiming He. “An empirical study of training self-supervised vision transformers”. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[34] Yuhua Chen, Wen Li, and Luc Van Gool. “Road: Reality oriented adaptation for semantic segmentation of urban scenes”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[35] Yuhua Chen et al. “Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[36] Mang Tik Chiu et al. “Agriculture-Vision: A Large Aerial Image Database for Agricultural Pattern Analysis”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[37] Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. “A downsampled variant of imagenet as an alternative to the cifar datasets”. In: *arXiv: 1707.08819* (2017).

[38] A. Coates, A. Ng, and H. Lee. “An Analysis of Single-Layer Networks in Unsupervised Feature Learning”. In: *International Conference on Artificial Intelligence and Statistics (ICAIS)*. 2011.

[39] Filipe R Cordeiro et al. “PropMix: Hard Sample Filtering and Proportional MixUp for Learning with Noisy Labels”. In: *arXiv: 2110.11809* (2021).

[40] Victor Guilherme Turrisi da Costa et al. “solo-learn: A Library of Self-supervised Methods for Visual Representation Learning”. In: *Journal of Machine Learning Research* (2022).

[41] Qiang Dai et al. “Crop leaf disease image super-resolution and identification with dual attention and topology fusion generative adversarial network”. In: *IEEE Access* (2020).

[42] Etienne David et al. “Global Wheat Head Detection (GWHD) dataset: a large and diverse dataset of high-resolution RGB-labelled images to develop and benchmark wheat head detection methods”. In: *Plant Phenomics* (2020).

[43] Y. Ding et al. “A Semi-Supervised Two-Stage Approach to Learning from Noisy Labels”. In: *IEEE Winter Conference on Applications of Computer Vision (WACV)*. 2018.

[44] C. Doersch, A. Gupta, and A. Efros. “Unsupervised Visual Representation Learning by Context Prediction”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2015.

[45] L. Dong-Hyun. “Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks”. In: *International Conference on Machine Learning Workshops (ICMLW)*. 2013.

[46] M. Donoser and H. Bischof. “Diffusion Processes for Retrieval Revisited”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013.

[47] Alexey Dosovitskiy et al. “Discriminative unsupervised feature learning with convolutional neural networks”. In: *Advances in neural information processing systems (NeurIPS)*. 2014.

[48] Alexey Dosovitskiy et al. “An image is worth 16x16 words: Transformers for image recognition at scale”. In: *International Conference on Learning Representations (ICLR)*. 2021.

[49] M. Douze et al. “Low-shot learning with large-scale diffusion”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[50] Lukas Drees et al. “Temporal prediction and evaluation of Brassica growth in the field using conditional generative adversarial networks”. In: *Computers and Electronics in Agriculture* (2021).

[51] M. Dusenberry et al. “Analyzing the role of model uncertainty for electronic health records”. In: *ACM Conference on Health, Inference, and Learning*. 2020.

[52] Debidatta Dwibedi et al. “With a little help from my friends: Nearest-neighbor contrastive learning of visual representations”. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[53] Michael Egan, Norann Galvin, and Deirdre Hennessy. “Incorporating white clover (Trifolium repens L.) into perennial ryegrass (Lolium perenne L.) swards receiving varying levels of nitrogen fertilizer:

Effects on milk and herbage production". In: *Journal of Dairy Science* 101.4 (2018), pp. 3412–3427.

[54] Aaron Etienne and Dharmendra Saraswat. "Machine learning approaches to automate weed detection by UAV based sensors". In: *Autonomous Air and Ground Sensing Systems for Agricultural Optimization and Phenotyping*. 2019.

[55] Mark Everingham and John Winn. "The pascal visual object classes challenge 2012 (voc2012) development kit". In: *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep* (2011).

[56] Chen Feng and Ioannis Patras. "Adaptive Soft Contrastive Learning". In: *International Conference on Pattern Recognition (ICPR)*. 2022.

[57] Z. Feng, C. Xu, and D. Tao. "Self-Supervised Representation Learning by Rotation Feature Decoupling". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[58] Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. "Exploring the Limits of Out-of-Distribution Detection". In: *Advances in neural information processing systems (NeurIPS)*. 2021.

[59] Kunihiko Fukushima and Sei Miyake. "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition". In: *Competition and cooperation in neural nets*. 1982.

[60] Quentin Garrido et al. "On the duality between contrastive and non-contrastive self-supervised learning". In: *arXiv: 2206.02574* (2022).

[61] S. Gidaris, P. Singh, and N. Komodakis. "Unsupervised Representation Learning by Predicting Image Rotations". In: *International Conference on Learning Representations (ICLR)*. 2018.

[62] Thomas Mosgaard Giselsson et al. “A public image database for benchmark of plant seedling classification algorithms”. In: *arXiv:1711.05458* (2017).

[63] J. Goldberger and E. Ben-Reuven. “Training deep neural-networks using a noise adaptation layer”. In: *International Conference on Learning Representations (ICLR)*. 2017.

[64] I. Goodfellow et al. “Generative adversarial nets”. In: *Advances in neural information processing systems (NeurIPS)*. 2014.

[65] Jean-Bastien Grill et al. “Bootstrap your own latent-a new approach to self-supervised learning”. In: *Advances in neural information processing systems (NeurIPS)*. 2020.

[66] C. Guo et al. “On calibration of modern neural networks.” In: *International Conference on Machine Learning (ICML)*. 2017.

[67] S. Guo et al. “CurriculumNet: Weakly Supervised Learning from Large-Scale Web Images”. In: *European Conference on Computer Vision (ECCV)*. 2018.

[68] B. Han et al. “Co-teaching: Robust training of deep neural networks with extremely noisy labels”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2018.

[69] Zane KJ Hartley and Andrew P French. “Domain Adaptation of Synthetic Images for Wheat Head Detection”. In: *Plants* (2021).

[70] Ali Hassani et al. “Escaping the big data paradigm with compact transformers”. In: *arXiv: 2104.05704* (2021).

[71] Sebastian Haug and Jörn Ostermann. "A crop/weed field image dataset for the evaluation of computer vision based precision agriculture tasks". In: *European Conference on Computer Vision*. 2014.

[72] Kaiming He et al. "Momentum contrast for unsupervised visual representation learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020.

[73] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. "Deep anomaly detection with outlier exposure". In: *International Conference on Learning Representations (ICLR)*. 2019.

[74] Dan Hendrycks et al. "Augmix: A simple data processing method to improve robustness and uncertainty". In: *International Conference on Learning Representations (ICLR)*. 2020.

[75] D Hennessy et al. "Using image analysis and machine learning to estimate sward clover content". In: *European Grassland Federation Symposium*. 2021.

[76] M Himstedt, T Fricke, and M Wachendorf. "The benefit of color information in digital image analysis for the estimation of legume contribution in legume–grass mixtures". In: *Crop Science* (2012).

[77] Fenner H Holman et al. "High throughput field phenotyping of wheat plant height and growth rate in field plot trials using UAV based remote sensing". In: *Remote Sensing* (2016).

[78] Junkai Huang et al. "Trash to Treasure: Harvesting OOD Data with Cross-Modal Matching for Open-Set Semi-Supervised Learning". In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[79] P. Husser, A. Mordvintsev, and D. Cremers. "Learning by Association - A versatile semi-supervised training method for neural networks". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[80] A. Iscen et al. "Efficient Diffusion on Region Manifolds: Recovering Small Objects with Compact CNN Representations". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[81] A. Iscen et al. "Label propagation for deep semi-supervised learning". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[82] N Islam et al. "Machine learning based approach for Weed Detection in Chilli field using RGB images". In: *International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*. 2020.

[83] Nahina Islam et al. "Early Weed Detection Using Image Processing and Machine Learning Techniques in an Australian Chilli Farm". In: *Agriculture* (2021).

[84] Phillip Isola et al. "Image-to-image translation with conditional adversarial networks". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

[85] Neal Jean, Sang Michael Xie, and Stefano Ermon. "Semi-supervised deep kernel learning: Regression with unlabeled data by minimizing predictive variance". In: *Advances in Neural Information Processing Systems (NeurIPS)* (2018).

[86] H. Jiabo et al. “Unsupervised Deep Learning by Neighbourhood Discovery”. In: *International Conference on Machine Learning (ICML)*. 2019.

[87] L. Jiang et al. “MentorNet: Learning Data-Driven Curriculum for Very Deep Neural Networks on Corrupted Labels”. In: *International Conference on Machine Learning (ICML)*. 2018.

[88] Lu Jiang et al. “Beyond Synthetic Noise: Deep Learning on Controlled Noisy Labels”. In: *International Conference on Machine Learning (ICML)*. 2020.

[89] Yu Jiang et al. “DeepSeedling: deep convolutional network and Kalman filter for plant seedling detection and counting in the field”. In: *Plant methods* (2019).

[90] Xiuliang Jin et al. “Estimates of plant density of wheat crops at emergence from very low altitude UAV imagery”. In: *Remote Sensing of Environment* (2017).

[91] Xiaotang Ju et al. “Nitrogen fertilization, soil nitrate accumulation, and policy recommendations in several agricultural regions of China”. In: *AMBIO: a Journal of the Human Environment* (2004).

[92] H. Kaiming et al. “Deep Residual Learning for Image Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[93] K. Kamnitsas et al. “Semi-Supervised Learning via Compact Latent Space Clustering”. In: *International Conference on Machine Learning (ICML)*. 2018.

[94] Henry J Kelley. “Gradient theory of optimal flight paths”. In: *Ars Journal* (1960).

[95] Saeed Khaki et al. “Wheatnet: A lightweight convolutional neural network for high-throughput image-based wheat head detection and counting”. In: *arXiv:2103.09408* (2021).

[96] Y. Kim et al. “NLNL: Negative Learning for Noisy Labels”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2019.

[97] A. Kolesnikov, X. Zhai, and L. Beyer. “Revisiting self-supervised visual representation learning”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[98] J Krause et al. “3D Object Representations for Fine-Grained Categorization”. In: *Workshop on 3D Representation and Recognition (3dRR-13)*. 2013.

[99] A. Krizhevsky and G. Hinton. *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto, 2009.

[100] A. Krizhevsky, I. Sutskever, and G. Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems (NeurIPS)*. 2012.

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

[102] Petre Lameski et al. “Weed detection dataset with RGB images taken under variable light conditions”. In: *International Conference on ICT Innovations*. 2017.

[103] Ya Le and Xuan Yang. “Tiny imagenet visual recognition challenge”. In: *CS 231N* (2015).

[104] Yann LeCun et al. “Backpropagation applied to handwritten zip code recognition”. In: *Neural computation* (1989).

[105] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* (1998).

[106] D. Lee and Y. Cheon. “Soft Labeling Affects Out-of-Distribution Detection of Deep Neural Networks”. In: *Workshop on International Conference on Machine Learning (ICMLW)*. 2020.

[107] Kibok Lee et al. “i-Mix: A Strategy for Regularizing Contrastive Representation Learning”. In: *International Conference on Learning Representations (ICLR)*. 2021.

[108] Kimin Lee et al. “Training confidence-calibrated classifiers for detecting out-of-distribution samples”. In: *International Conference on Learning Representations (ICLR)*. 2018.

[109] Michael A Lefsky et al. “Lidar remote sensing for ecosystem studies: Lidar, an emerging remote sensing technology that directly measures the three-dimensional distribution of plant canopies, can accurately estimate vegetation structural attributes and should be of particular interest to forest, landscape, and global ecologists”. In: *BioScience* (2002).

[110] Yu-Feng Li, Han-Wen Zha, and Zhi-Hua Zhou. “Learning safe prediction for semi-supervised regression”. In: *AAAI Conference on Artificial Intelligence*. 2017.

[111] J. Li, R. Socher, and S.C.H. Hoi. “DivideMix: Learning with Noisy Labels as Semi-supervised Learning”. In: *International Conference on Learning Representations (ICLR)*. 2020.

[112] Jie Li, Junjie Jia, and Donglai Xu. “Unsupervised representation learning of image-based plant disease with deep convolutional generative adversarial networks”. In: *2018 37th Chinese control conference (CCC)*. 2018.

[113] Junnan Li, Caiming Xiong, and Steven CH Hoi. “Learning from noisy data with robust representation learning”. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[114] Junnan Li et al. “Learning to learn from noisy labeled data”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[115] W. Li et al. “WebVision Database: Visual Learning and Understanding from Web Data”. In: *arXiv: 1708.02862* (2017).

[116] Y. Li, L. Liu, and R. Tan. “Certainty-Driven Consistency Loss for Semi-supervised Learning”. In: *arXiv: 1901.05657* (2019).

[117] Yanghao Li et al. “Revisiting batch normalization for practical domain adaptation”. In: *International Conference on Learning Representations Worksop (ICLRW)*. 2017.

[118] Tsung-Yi Lin et al. “Microsoft coco: Common objects in context”. In: *European conference on computer vision (ECCV)*. 2014.

[119] S. Liu et al. “Early-Learning Regularization Prevents Memorization of Noisy Labels”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2020.

[120] Tao Liu et al. “Evaluation of seed emergence uniformity of mechanically sown wheat with UAV RGB imagery”. In: *Remote Sensing* (2017).

[121] Ze Liu et al. “Swin transformer: Hierarchical vision transformer using shifted windows”. In: *IEEE/CVF International Conference on Computer Vision (CVPR)*. 2021.

[122] Zhuang Liu et al. “A convnet for the 2020s”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022.

[123] M. Lukasik et al. “Does label smoothing mitigate label noise?” In: *International Conference on Machine Learning (ICML)*. 2020.

[124] Yawei Luo et al. “Significance-aware information bottleneck for domain adaptive semantic segmentation”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2019.

[125] Xingjun Ma et al. “Normalized loss functions for deep learning with noisy labels”. In: *International conference on machine learning (ICML)*. 2020.

[126] Y. Mang et al. “Unsupervised Embedding Learning via Invariant and Spreading Instance Feature”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[127] Cheryl L McCarthy, Nigel H Hancock, and Steven R Raine. “Applied machine vision of plants: a review with implications for field deployment in automated farming operations”. In: *Intelligent Service Robotics* (2010).

[128] Warren S McCulloch and Walter Pitts. “A logical calculus of the ideas immanent in nervous activity”. In: *The bulletin of mathematical biophysics* (1943).

[129] Andres Milioto, Philipp Lottes, and Cyrill Stachniss. “Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in CNNs”. In: *IEEE international conference on robotics and automation (ICRA)*. 2018.

[130] Massimo Minervini et al. “Finely-grained annotated datasets for image-based plant phenotyping”. In: *Pattern recognition letters* (2016).

[131] T. Miyato et al. “Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2017).

[132] Anders K Mortensen et al. “Preliminary results of clover and grass coverage and total dry matter estimation in clover-grass crops using image analysis”. In: *Journal of Imaging* (2017).

[133] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. “When does label smoothing help?” In: *Advances in neural information processing systems (NeurIPS)* (2019).

[134] F Nájera et al. “Evaluation of soil fertility and fertilisation practices for irrigated maize (Zea mays L.) under Mediterranean conditions in central Chile”. In: *Journal of soil science and plant nutrition* (2015).

[135] Badri Narayanan et al. “Extracting pasture phenotype and biomass percentages using weakly supervised multi-target deep learning on

a small dataset”. In: *Irish Machine Vision and Image Processing conference*. 2020.

[136] Haseeb Nazki et al. “Image-to-image translation with GAN for synthetic data augmentation in plant disease datasets”. In: *Smart Media Journal* (2019).

[137] Haseeb Nazki et al. “Unsupervised image translation using adversarial networks for improved plant disease recognition”. In: *Computers and Electronics in Agriculture* (2020).

[138] Andrew Y Ng, Michael I Jordan, and Yair Weiss. “On spectral clustering: Analysis and an algorithm”. In: *Advances in neural information processing systems (NeurIPS)*. 2002.

[139] Ben Niu et al. “Single image super-resolution via a holistic attention network”. In: *European conference on computer vision (ECCV)*. 2020.

[140] J. Nixon et al. “Measuring Calibration in Deep Learning.” In: *IEEE Workshop on Computer Vision and Pattern Recognition (CVPRW)*. 2019.

[141] Mehdi Noroozi and Paolo Favaro. “Unsupervised learning of visual representations by solving jigsaw puzzles”. In: *European Conference on Computer Vision (ECCV)*. 2016.

[142] Daniel Nyfeler et al. “Strong mixture effects among four species in fertilized agricultural grassland led to persistent and consistent transgressive overyielding”. In: *Journal of Applied Ecology* (2009).

[143] D. Ortego et al. “Towards Robust Learning with Different Label Noise Distributions”. In: *arXiv: 1912.08741* (2019).

[144] D. Ortego et al. “Towards Robust Learning with Different Label Noise Distributions”. In: *International Conference on Pattern Recognition (ICPR)*. 2020.

[145] Diego Ortego et al. “Multi-Objective Interpolation Training for Robustness to Label Noise”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021.

[146] Diego Ortego et al. “Towards robust learning with different label noise distributions”. In: *International Conference on Pattern Recognition (ICPR)*. 2021.

[147] Ajinkya Paikekari et al. “Weed detection using image processing”. In: *International Research Journal of Engineering and Technology (IRJET)* (2016).

[148] Taesung Park et al. “Contrastive learning for unpaired image-to-image translation”. In: *European Conference on Computer Vision (ECCV)*. Springer. 2020, pp. 319–345.

[149] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2019.

[150] G. Patrini et al. “Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[151] Abriti Paul et al. “A review on agricultural advancement based on computer vision and machine learning”. In: *Emerging technology in modelling and graphics* (2020).

[152] José Luis Pech-Pacheco et al. “Diatom autofocusing in brightfield microscopy: a comparative study”. In: *International Conference on Pattern Recognition (ICPR)*. 2000.

[153] Camilo Andra Pulido-Rojas, Manuel Alejandro Molina-Villa, and Leonardo Enrique Solaque-GuzmÃ¡n. “Machine vision system for weed detection using image filtering in vegetables crops”. In: *Revista Facultad de IngenierÃa Universidad de Antioquia* (2016).

[154] F. Radenovic, G. Tolias, and O. Chum. “Fine-tuning CNN Image Retrieval with No Human Annotation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2018).

[155] Alec Radford et al. “Learning transferable visual models from natural language supervision”. In: *International Conference on Machine Learning (ICML)*. 2021.

[156] M. Raghu et al. “Direct uncertainty prediction for medical second opinions”. In: *International Conference on Machine Learning (ICML)*. 2019.

[157] A. Rasmus et al. “Semi-Supervised Learning with Ladder Network”. In: *Advances in Neural Information Processing Systems (NeuRIPS)*. 2015.

[158] S-A. Rebuffi et al. “Semi-Supervised Learning with Scarce Annotations”. In: *arXiv: 1905.08845* (2019).

[159] S. Reed et al. “Training deep neural networks on noisy labels with bootstrapping”. In: *International Conference on Learning Representations (ICLR)*. 2015.

[160] Zhongzheng Ren, Raymond Yeh, and Alexander Schwing. “Not all unlabeled data are equal: Learning to weight data in semi-supervised learning”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2020.

[161] Stephan R Richter et al. “Playing for data: Ground truth from computer games”. In: *European conference on computer vision*. 2016.

[162] Joshua Robinson et al. “Contrastive learning with hard negative samples”. In: *International Conference on Learning Representations (ICLR)*. 2020.

[163] G. Ros et al. “The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[164] Inkyu Sa et al. “Deepfruits: A fruit detection system using deep neural networks”. In: *Sensors* (2016).

[165] Ragav Sachdeva et al. “EvidentialMix: Learning with Combined Open-set and Closed-set Noisy Labels”. In: *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2020.

[166] Ragav Sachdeva et al. “ScanMix: Learning from Severe Label Noise via Semantic Clustering and Semi-Supervised Learning”. In: *arXiv: 2103.11395* (2021).

[167] Swami Sankaranarayanan et al. “Unsupervised domain adaptation for semantic segmentation with gans”. In: *arXiv: 1711.06969* (2017).

[168] Christoph Schuhmann et al. “Laion-400m: Open dataset of clip-filtered 400 million image-text pairs”. In: *Advances in neural information processing systems (NeurIPS)*. 2022.

[169] Murat Sensoy, Lance Kaplan, and Melih Kandemir. “Evidential deep learning to quantify classification uncertainty”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2018.

[170] Yanyao Shen and Sujay Sanghavi. “Learning with bad training data via iterative trimmed loss minimization”. In: *International Conference on Machine Learning (ICML)*. 2019.

[171] Jianbo Shi and Jitendra Malik. “Normalized cuts and image segmentation”. In: *IEEE Transactions on pattern analysis and machine intelligence (TPAMI)* (2000).

[172] Ashish Shrivastava et al. “Learning from simulated and unsupervised images through adversarial training”. In: *IEEE conference on computer vision and pattern recognition (CVPR)*. 2017.

[173] K. Simonyan and A. Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *International Conference on Learning Representations (ICLR)*. 2015.

[174] Soren Skovsen et al. “The GrassClover image dataset for semantic and hierarchical species understanding in agriculture”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2019.

[175] Søren Skovsen et al. “Predicting dry matter composition of grass clover leys using data simulation and camera-based segmentation of field canopies into white clover, red clover, grass and weeds”. In: *International Conference on Precision Agriculture*. 2018.

[176] Søren Kelstrup Skovsen et al. “Robust species distribution mapping of crop mixtures using color images and convolutional neural networks”. In: *Sensors* (2021).

[177] Karen Søegaard. “Nitrogen fertilization of grass/clover swards under cutting or grazing by dairy cows”. In: *Acta Agriculturae Scandinavica Section B–Soil and Plant Science* (2009).

[178] K. Sohn et al. “FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence”. In: *arXiv: 2001.07685* (2020).

[179] Kihyuk Sohn. “Improved deep metric learning with multi-class n-pair loss objective”. In: *Advances in neural information processing systems (NeurIPS)*. 2016.

[180] H. Song, M. Kim, and J.-G. Lee. “SELFIE: Refurbishing Unclean Samples for Robust Deep Learning”. In: *International Conference on Machine Learning (ICML)*. 2019.

[181] H. Song et al. “Learning from noisy labels with deep neural networks: A survey”. In: *arXiv: 2007.08199* (2020).

[182] Chen Sun et al. “Revisiting unreasonable effectiveness of data in deep learning era”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2017.

[183] Zeren Sun et al. “Webly Supervised Fine-Grained Recognition: Benchmark Datasets and An Approach”. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[184] C. Szegedy et al. “Going Deeper with Convolutions”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2015.

[185] C. Szegedy et al. “Inception-v4, inception-resnet and the impact of residual connections on learning”. In: *Association for the Advancement of Artificial Intelligence (AAAI)*. 2016.

[186] M. Szummer and J. Tommi. “Partially labeled classification with Markov random walks”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2002.

[187] Mingxing T. and Quoc L. “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”. In: *International Conference on Machine Learning (ICML)*. 2019.

[188] D. Tanaka et al. “Joint Optimization Framework for Learning with Noisy Labels”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[189] Jing-Lei Tang et al. “Weed detection using image processing under different illumination for site-specific areas spraying”. In: *Computers and Electronics in Agriculture* (2016).

[190] A. Tarvainen and H. Valpola. “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2017.

[191] Joshua B Tenenbaum, Vin de Silva, and John C Langford. “A global geometric framework for nonlinear dimensionality reduction”. In: *Science* (2000).

[192] B. Thomee et al. “The New Data and New Challenges in Multimedia Research”. In: *arXiv: 1503.01817* (2015).

[193] S. Thulasidasan et al. “On mixup training: Improved calibration and predictive uncertainty for deep neural networks”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2019.

[194] Hongkun Tian et al. “Computer vision technology in agricultural automation. A review”. In: *Information Processing in Agriculture* (2020).

[195] Mohan Timilsina et al. “Semi-supervised regression using diffusion on graphs”. In: *Applied Soft Computing* 104 (2021), p. 107188.

[196] Marco Toldo et al. “Unsupervised domain adaptation in semantic segmentation: a review”. In: *Technologies* (2020).

[197] G. Tolias, Y. Avrithis, and H. Jégou. “To Aggregate or Not to aggregate: Selective Match Kernels for Image Search”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2013.

[198] M. Toneva et al. “An empirical study of example forgetting during deep neural network learning”. In: *International Conference on Learning Representations (ICLR)*. 2019.

[199] Muhammad Toseef and Malik Jahan Khan. “An intelligent mobile application for diagnosis of crop diseases in Pakistan using fuzzy inference system”. In: *Computers and Electronics in Agriculture* (2018).

[200] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems (NeurIPS)*. 2017.

[201] V. Verma et al. “Interpolation Consistency Training for Semi-Supervised Learning”. In: *International Joint Conferences on Artificial Intelligence (IJCAI)*. 2019.

[202] O. Vinyals et al. “Matching Networks for One Shot Learning”. In: *Advances in Neural Information Processing Systems (NeuRIPS)*. 2016.

[203] Tuan-Hung Vu et al. “Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[204] N. Vyas, S. Saxena, and T. Voice. “Learning Soft Labels via Meta Learning”. In: *arXiv: 2009.09496* (2020).

[205] Guan Wang, Yu Sun, and Jianxin Wang. “Automatic image-based plant disease severity estimation using deep learning”. In: *Computational intelligence and neuroscience* (2017).

[206] Tongzhou Wang and Phillip Isola. “Understanding contrastive representation learning through alignment and uniformity on the hypersphere”. In: *International Conference on Machine Learning (ICLR)*. 2020.

[207] Xiao Wang et al. “EnAET: A self-trained framework for semi-supervised and supervised learning with ensemble transformations”. In: *IEEE Transactions on Image Processing* (2020).

[208] Y. Wang et al. “Iterative Learning With Open-Set Noisy Labels”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[209] Y. Wang et al. “Symmetric cross entropy for robust learning with noisy labels”. In: *IEEE International Conference on Computer Vision (ECCV)*. 2019.

[210] Yisen Wang et al. “Symmetric cross entropy for robust learning with noisy labels”. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019.

[211] Michelle Watt et al. “Phenotyping: new windows into the plant for breeders”. In: *Annual review of plant biology* (2020).

[212] Z. Wu et al. “Unsupervised feature learning via non-parametric instance discrimination”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[213] T. Xiao et al. “Learning from massive noisy labeled data for image classification”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

[214] Q. Xie et al. “Unsupervised Data Augmentation for Consistency Training”. In: *arXiv: 1904.12848* (2019).

[215] Youjiang Xu et al. “Faster meta update strategy for noise-robust deep learning”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021.

[216] Yazhou Yao et al. “Jo-SRC: A Contrastive Approach for Combating Noisy Labels”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021.

[217] Kun Yi and Jianxin Wu. “Probabilistic end-to-end noise correction for learning with noisy labels”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[218] Li Yi et al. “On Learning Contrastive Representations for Learning With Noisy Labels”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022.

[219] Qing Yu and Kiyoharu Aizawa. "Unsupervised out-of-distribution detection by maximum classifier discrepancy". In: *IEEE International Conference on Computer Vision (ICCV)*. 2019.

[220] Xingrui Yu et al. "How does disagreement help generalization against label corruption?" In: *International Conference on Machine Learning (ICML)*. 2019.

[221] S. Zagoruyko and N. Komodakis. "Wide residual networks". In: *arXiv: 1605.07146* (2016).

[222] Jure Zbontar et al. "Barlow twins: Self-supervised learning via redundancy reduction". In: *International Conference on Machine Learning (ICML)*. 2021.

[223] Xiaohua Zhai et al. "S4l: Self-supervised semi-supervised learning". In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019.

[224] Zhaoyu Zhai et al. "A mission planning approach for precision farming systems based on multi-objective optimization". In: *Sensors* (2018).

[225] Bowen Zhang et al. "Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2021.

[226] C. Zhang et al. "Understanding deep learning requires re-thinking generalization". In: *International Conference on Learning Representations (ICLR)*. 2017.

[227] H. Zhang et al. "mixup: Beyond Empirical Risk Minimization". In: *International Conference on Learning Representations (ICLR)*. 2018.

[228] L. Zhang et al. “Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[229] R. Zhang, P. Isola, and A. A Efros. “Colorful image colorization”. In: *European Conference on Computer Vision (ECCV)*. 2016.

[230] Bolei Zhou et al. “Places: A 10 million Image Database for Scene Recognition”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017).

[231] D. Zhou et al. “Learning with Local and Global Consistency”. In: *International Conference on Neural Information Processing Systems (NeurIPS)*. 2003.

[232] Zhi-Hua Zhou, Ming Li, et al. “Semi-supervised regression with co-training.” In: *International Joint Conference on Artifical Intelligence (IJCAI)*. 2005.

[233] J.-Y. Zhu et al. “Unpaired image-to-image translation using cycle-consistent adversarial networks”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2017.

[234] Yezi Zhu et al. “Data Augmentation using Conditional Generative Adversarial Networks for Leaf Counting in Arabidopsis Plants.” In: *British Machine Vision Conference (BMVC)*. 2018.

Chapter 9

Appendix

9.1 Deep learning

9.1.1 Deep neural networks

Deep learning for computer vision is a discipline of Machine Learning whereby a computer learns visual features and consequently reduces the need for humans to handcraft attributes for the classification, detection, or segmentation of visual objects. Although the foundation theory for deep learning was laid out in the middle of the twentieth century [59, 94, 104, 128], low computing capacities available at the time limited the application of the algorithms. Before the standardization of neural networks for computer vision, machine learning approaches relied on handcrafted features where researchers would manually specify unique attributes for the classification of images. The attributes could, for example, include the expected position of the ears, eyes, or limbs for an animal classification task. The detection of the different features would then be used by a classification model such as a Support Vector Machine to perform the final classification choice given the activations of the handcrafted features. A first successful milestone that laid the foundations for modern deep learning is the entry submitted by Alex Krizhevsky and Geoffrey Hinton to the ILSVRC2012 challenge, where the proposed neural network architecture AlexNet decreased the classification error on the challenge by more than 10 points when compared with state-of-the-art approaches at the time [101]. This entry cemented the relevance of neural architectures for solving computer vision tasks.

The dominant advantage of training a neural network is that it removes the need for human experts to define important attributes and instead allows patterns to be learned directly from the training data. To do so, a deep neural network learns a feature extractor composed of successive layers of neurons that linearly combine the set of input pixels passed on to them. To drastically improve the representation power of the neural network, the linear combination is followed by a non-linearity or activation function. A classic fully connected neural network architecture (multi-layer perceptron) is composed of: an input layer, which extracts the initial feature from the input image; a set of hidden layers built on top of the input layer that combine the extracted features in a non-linear fashion; a classification head that maps the feature activations to class predictions. To learn the parameters in each layer of the neural network, updates are computed using a gradient descent process, derived from an error function between the target prediction and actual prediction. Gradient descent is explained in more details in section 9.1.2. A common justification for the neuron structure in neural network is a coarse bio-inspiration from synapse connections in the brain [23]. Figure 9.1 illustrates a simple comparison between a neuron and a simple fully connected architecture.

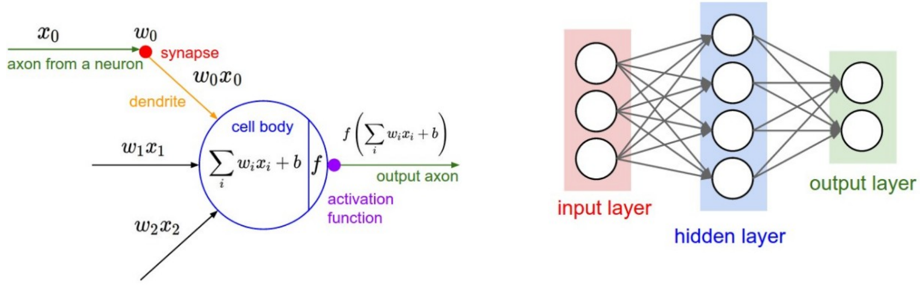

Figure 9.1: Structure of a neuron (left) and of a fully connected neural network with a hidden layer (right). Figure from `http://cs231n.github.io/neural-networks-1/`.

9.1.2 Gradient descent

We develop here the gradient descent process used to iteratively update parameters in neural networks. The gradient has to be derived from a cost function. A commonly used loss function for image classification (the main research field studied in this thesis) is the cross entropy loss. Considering a dataset composed of N labeled samples $\mathcal{D} = \{x_i, y_i\}^N$, the cross entropy loss is computed between the prediction of the network over a training sample x_i (image) that we denote $\hat{y}_i$ and the ground truth label y_i. C is the number of classes in the dataset and the size of both vectors. The objective is to learn parameters that minimize the cross-entropy loss over all images in the training dataset:

$$l_{ce} = \frac{1}{N} \sum_{i=1}^{N} -y_i^T \log(\hat{y}_i), \tag{9.1}$$

where the $\log$ operation is applied element wise. Once the network's error is quantified by the cross-entropy loss function, the gradient descent algorithm is then applied to update the parameters. Given θ_t the network's parameters used to compute $\hat{y}_i$ at iteration t, an update is computed for θ_{t+1} in the opposite direction of the gradient of l_{ce}. The strength of the update is modulated by a hyperparameter α (learning rate):

$$\theta_{t+1} = \theta_t - \alpha \nabla_{\theta_t} l_{ce}, \tag{9.2}$$

where l_{ce} the cross-entropy loss is a function of the network's parameters θ_t and ∇_θ represent the gradient operation with regards to the networks parameters θ. To update the network parameters in every successive layer θ^i, the chain rule is used:

$$\frac{\partial l_{ce}}{\partial \theta} = \frac{\partial l_{ce}}{\partial \theta^{i+1}} \frac{\partial \theta^{i+1}}{\partial \theta^{i}} \frac{\partial \theta^{i}}{\partial \theta^{i-1}} \frac{\partial \theta^{i-1}}{\partial \theta}. \tag{9.3}$$

The chain rule allows iterative propagation of the gradient update through the network, starting from the final layer $\frac{\partial l_{ce}}{\partial \theta^{i+1}}$ all the way to the first.

The dominant limitation of gradient descent is that each gradient update step has to be averaged over the full dataset. This is impossible in practice because large datasets coupled with the large amounts of parameters of neural networks would lead to memory limitations. The commonly used alternative is named Stochastic Gradient Descent, which approximates the gradient descent step by performing it on a randomly selected subset of the dataset. This practice is commonly named mini-batching and the size of the mini-batch can be set accordingly with hardware limitations. Finally, regarding the initial network parameters a random initialization is performed.

An important hyper-parameter when computing the weight update is the learning rate (α parameter of Equation 9.2). This learning rate parameter can be thought of the step size taken in the opposite direction of the gradient. Taking large steps at the beginning of training can be useful to accelerate training and to avoid local minima but it is common to reduce the step size later in training to help convergence. The common practice is to set the learning rate to a high value initially and to reduce it in a step-wise fashion during training.

9.1.3 Convolution operation for computer vision

Although multi-layer perceptron (MLP) architectures (see section 9.1.1) can be applied on simple datasets composed of limited numbers of input variables, the number of learnable parameters will quickly expand when applied on large and complex objects such as images. When considering the application of a simple MLP to a RGB image of $224 \times 224 \times 3$ pixels in size (a common input image size for neural networks in the ILSVRC2012 [100] challenge), the fully connected nature of the architecture requires $224 \times 224 \times 3 = 150,528$ connections for each neuron in the input layer alone, resulting in very large matrix multiplications and rendering the computation difficult in a reasonable time. Other MLP limitations for image processing include: the absence of spatial coherence as the order of the connections between successive layers is unimportant (an image with shuffled pixels would not be a problem for a fully connected architecture); the impossibility to adapt to images of different resolutions; overfitting to the training data because of the large number of parameters.

To address this issue, Convolutional Neural Networks [105] (CNNs)

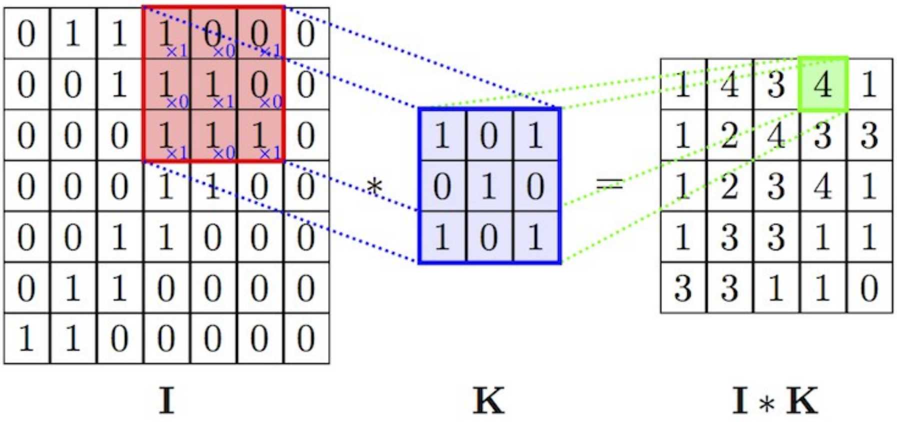

Figure 9.2: Visualization of the convolution operation in convolutional neural networks with a kernel size of 3. Figure from `http://jenslaufer.com/`.

were proposed to greatly reduce the high parameterization characteristic of fully connected neural networks when applied to image processing. In the case of a CNN, the hidden layers are replaced by convolution layers. A convolution layer is a collection of learnable 2D filters of the same size that are applied on every input pixel and their spacial neighbours in the image. This computation strategy reduces the number of connections between successive representations to the number of pixels covered by the convolution filter: typically 7×7 or 3×3 multiplied by the depth (number of filters) of the previous representation multiplied the depth of the output representations (typically between 1 and 512). The convolution process is a classic filtering operation where visual features activating the filters anywhere in the image will be detected and passed on to the following layer in order to be refined. The convolution operation for neural networks is represented in Figure 9.2. Other advantages of the convolution architecture includes localized spatial coherence and the adaptability to images of different sizes.

To understand the complexity of the visual features learned in deeper

CNN layers, an interesting experiment is to generate an input image that maximally activates a given filter in the trained neural network. This is accomplished using an inverted gradient process where, given a weight-frozen CNN trained on an image dataset, the input image is updated using gradient ascent (as opposed to descent) to maximize the activation of a learned filter. When observing the images obtained for filters in the first layer these appear very simple such as color or direction detection. The deeper the layer the more complex the patterns that are created. Figure 9.3 gives examples of generated images from a VGG16 [173] architecture trained on ImageNet. Other visualization processes have been developed where gradient ascent is used to generate an image that activates a specific class (maximize the activation of a given class in the last fully connected layer) or, where given an input image and a target class, one can look at which part of the image maximally activates the filters responsible for the target class prediction (feature activation maps).

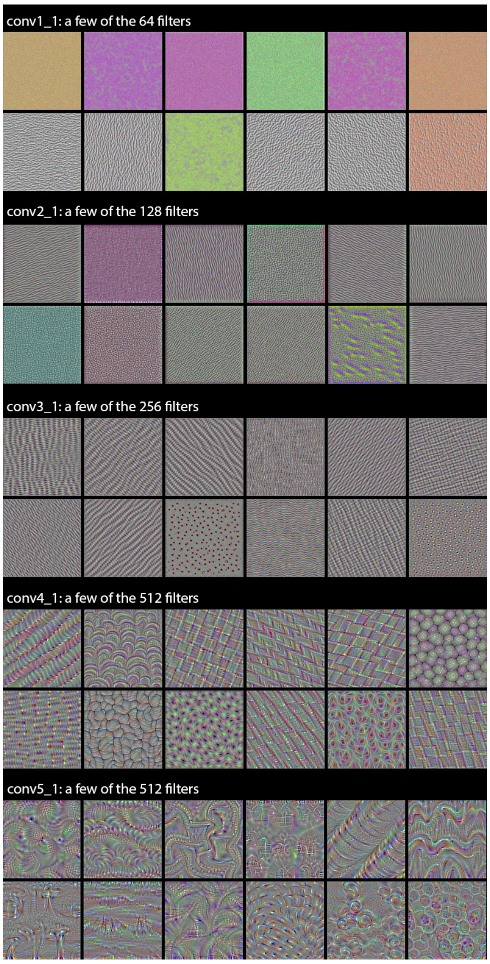
conv1_1: a few of the 64 filters
conv2_1: a few of the 128 filters
conv3_1: a few of the 256 filters
conv4_1: a few of the 512 filters
conv5_1: a few of the 512 filters

Figure 9.3: Visualizing images that strongly activate a filter in a given layer of a VGG16 trained on ImageNet. Source `https://blog.keras.io/how-convolutional-neural-networks-see-the-world.html`