Few-shot learning for image classification and object detection

by

Edward Zimudzi

submitted in accordance with the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in the subject

COMPUTER SCIENCE

at the

UNIVERSITY OF SOUTH AFRICA

SUPERVISOR: PROF I D SANDERS

NOVEMBER 2021

Preface

Part of this work has been published as:

1. Edward Zimudzi, Ian Sanders, Nicholas Rollings & Christian Omlin (2018) Segmenting mangrove ecosystems drone images using SLIC superpixels, *Geocarto International*, Taylor & Francis. DOI: 10.1080/10106049.2018.1497093. Available at https://www.tandfonline.com/doi/full/10.1080/10106049.2018.1497093

2. Edward Zimudzi, Ian Sanders, Nicholas Rollings & Christian Omlin (2019). Remote sensing of Mangroves Using Unmanned Aerial Vehicles: Current state and future directions, *Journal of Spatial Science*, Taylor & Francis.
   DOI: 10.1080/14498596.2019.1627252. Available at
   https://doi.org/10.1080/14498596.2019.1627252.

This thesis is dedicated to my late father Mr. Fidelis Runouya (FR) Zimudzi who taught me that to achieve something in life requires complete confidence, trust and belief in yourself, firmness of purpose, perseverance, dedications, and dedication

Declaration

"I declare that **Few-shot Learning for Image Classification and Object Detection** is my own work and that all the sources that I have used or quoted have been indicated and acknowledged by means of complete references.

I further declare that I submitted the thesis to originality checking software and that it falls within the accepted requirements for originality.

I further declare that I have not previously submitted this work, or part of it, for examination at Unisa for another qualification or at any other higher education institution."

Signature: _____

Edward Zimudzi

Student Number - 51501813

Acknowledgements

First and foremost I want to express my sincere gratitude and thanks to my thesis supervisor Professor Ian Douglas Sanders. I feel he has been a great mentor, a collaborator, and a friend. I thank him for offering me the opportunity to do a PhD on such an interesting and challenging deep learning and computer vision topic and providing me the platform to study for this thesis. This thesis would not have been easily completed without his commitment and diligent efforts over the years, which not only influenced the content and layout of the thesis but also the language in which it has been written. He not only showed how to exhaustively explore different methodologies and analyse results, but always encouraged me for the need for flawlessness and perfectness not only when performing research but also when communicating the research results. I would promptly receive detailed reports, corrections and comments for all the work I submitted to him despite the amount of engagements at his other end, reflecting his dedication and responsibility. I am also grateful to Professor Christian W. Omlin who was co-supervising during the initial stages of my studies, and always encouraged my resourcefulness, drive, and inventiveness to make the PhD research more connected to applications in daily life. He has also been very supportive through writing journal articles and initial contact of Nicholas Rollings at the University of the South Pacific for the provision of data from The Suva Foreshore in Fiji. The assistance received from them both professionally and individually was tremendous.

I would like to thank all my friends and colleagues for the encouragement and for

making DMSE an enjoyable and fascinating place to work in. High regard and appreciation to my office floor colleagues, Salome Maemo, Paul Algebra, Ignatius, Kay, Motswiri, Kgomotso, MmaKesianye, Lesego, Alex, Antony, Robert, Shanah, Jimmy, Ethel, Pitso, Lanka, Matongo, Polelo, and Nyepetsi for their endurance with my actions during these past seven years. I shared many interesting discussions and work time fun with them.

I cannot end without thanking my mother and late father for all the encouragement and vision, and my brothers, Clemence, Febias (the late), Dzingai, Pepukai and Ernest, and sister Evelyn for their absolute confidence in me. My final words go to Rufaro G., my wife and children, Tinotenda Evelyn, Ruvimbo Melissa and Tafara Ernest. In this line of work families suffer the most. Thank you for endurance with my late hours, many spoiled weekends and vacations, and above all for staying by my side.

Abstract

Deep learning has successfully been applied in computer vision, including in image classification, object recognition and detection, and in image segmentation in applications such as remote sensing, scene understanding, autonomous driving, medical image analysis, robotics and video surveillance. The drawback of the majority of current approaches is that they demand huge quantities of annotated training data to produce results, and they use quite expensive computing resources. Data annotation is usually an expensive and tedious task. On the other hand, data can be rare or difficult to gather for some reasons, including for safety and ethical issues. Moreover, a deep learning model trained successfully for a specific task cannot be directly deployed for another task in another domain. It is therefore essential to develop models that can learn from few annotated samples of training data like humans do. Few-shot learning addresses the problem of closing the gap into deep learning models that learn from huge annotated datasets and humans in the challenging task of learning from few examples. The aim of this thesis is to propose novel methods in deep learning image processing that optimize the model's ability to detect and recognise new object instances using few labelled data.

We present several novel methods that tackle the problems of image classification, object detection, self-supervised knowledge distillation, and panoptic segmentation in few-shot learning settings. Even though multiple computer vision themes can be identified throughout this work, the most important is the limited data regime taken into account. We consider the few-shot learning setting where tasks associated with their support and

query test data are received and trained in episodes. We introduce a novel few-shot meta-learning classification model that consists of multiple learners supervised by a central controller to control a feature extraction and meta-learning for integrated inference and generalisation. Secondly, we introduce an approach for few-shot object detection that meta-learns object localisation and classification by eliminating region-wise prediction, and encoding support images and query images simultaneously into class-specific feature representations that automatically enters into a class-agnostic decoder to generate output predictions for the categories known beforehand. We also introduce a fully convolutional model for panoptic segmentation in few-shot settings that encodes each instance into a specific kernel and generates a prediction by convolutions directly, thereby predicting both instance objects and background stuff together. In this way, instance-aware and semantically consistent properties for object instances and their background can be respectively satisfied in a unified workflow. Finally, we introduce a two-stage knowledge distillation model that maximises the entropy of the feature embeddings of images using a self-supervised auxiliary loss. Experiments on some public few-shot learning benchmark datasets such as miniImageNet, Omniglot, COCO-20^i and Mapillary Vistas demonstrate the effectiveness of the proposed methods for few-shot learning in computer vision.

Keywords

Few-shot learning, image classification, object detection, knowledge distillation, panoptic segmentation, deep neural networks, meta-learning, metric learning, image processing

Thesis Summary

Thesis Title: Few-Shot Learning for Image Classification and Object Detection

This thesis presents and optimise several novel models that tackle computer vision problems of image classification, object detection, self-supervised knowledge distillation, and panoptic segmentation in few-shot learning settings. Few-shot learning aims to close the gap between deep learning models that learn from huge, annotated datasets and humans in the challenging task of learning from a few annotated examples. Even though multiple computer vision themes can be identified throughout this work, the most important is the limited data regime considered. We consider the few-shot learning settings where computer vision tasks with limited data associated with their support and query test data are received and trained in episodes. We first introduce a novel few-shot meta-learning classification model that consists of multiple learners supervised by a central controller to control feature extraction and meta-learning for integrated inference and generalisation. Secondly, we introduce an approach for few-shot object detection that detects and recognises new object instances by meta-learning object localisation and classification in a unified manner by eliminating region-wise prediction and encoding both support images and query images into category-specific features that then feeds into a category-agnostic decoder to generate predictions for the specific categories. We also introduce a fully convolutional model for panoptic segmentation in few-shot settings that encodes each instance into a specific kernel and generates a prediction by convolutions directly, thereby predicting both instance objects and background stuff together. In this

way, instance-aware and semantically consistent properties for object instances and their background can be respectively satisfied in a unified workflow. Finally, we introduce a two-stage knowledge distillation model that maximises the entropy of the feature embeddings of images using a self-supervised auxiliary loss. Experiments on some public few-shot learning benchmark datasets such as miniImageNet, Omniglot, CIFAR-FS and Oxford Flowers102 for image classification, Pascal 5^i and COCO-20^i for object detection, and Mapillary Vistas for panoptic segmentation demonstrate the performance parameters and the effectiveness of the proposed methods for few-shot learning. This thesis aims to close the gap between conventional deep learning and human learning by creating computer vision systems that learn from a few examples of image data.

Keywords Few-shot learning, image classification, object detection, knowledge distillation, panoptic segmentation, deep neural networks, meta-learning, metric learning, image processing

IsiZulu

Isihloko sendaba ende: Indlela yokufunda yomshini yokuhlukaniswa kwemifanekiso nokutholwa kwento

Le ndaba ende yethula futhi ithuthukise izifanekiso zamanoveli amaningana abhekana nezinkinga zokubono zekhompyutha zokuhlukaniswa kwemifanekiso, ukutholwa kwezinto, ukuhluzwa kolwazi oluzigadile kanye nendlela yokuhlukanisa umfanekiso osetshenziselwa imisebenzi yokubona yekhompyutha eqoqweni lendlela yokufunda yomshini. Indlela yokufunda yomshini ihlose ukuvala igebe phakathi kwezifanekiso zokufunda ezijulile ezifunda eqoqweni elikhulu lemininingwane yolwazi ehlobene, ezinezichasiselo nakubantu emsebenzini oyinselele wokufunda ezibonelweni ezimbalwa ezinezichasiselo. Ngisho noma izindikimba eziningi zokubono zekhompyutha zingabonakala kuwo wonke lo msebenzi, okubaluleke kakhulu uhlelo lwemininingwane olulinganiselwe olucatshangelwayo. Sicabangela iqoqo lendlela yokufunda yomshini lapho imisebenzi yokubona yekhompuyutha inemininingwane elinganiselwe ehlotshaniswa nokusekelwa kwayo kanye neminingwane yokuhlola imibuzo iyatholwa futhi iqeqeshwe ngeziqephu. Okokuqala sethula isifanekiso sokuhlukaniswa kokufunda ukufunda kwenoveli okumbalwa okuhlanganisa abafundi abaningi abagadwe yisilawuli esimaphakathi ukuze kulawulwe ukukhishwa kwesici kanye nokufunda ukufunda ukuze kufinyelelwe ekucabangeni okudidiyelwe kanye nokujwayelekile, Okwesibili, sethula indlela yokuthola izinto zendlela yokufunda yomshini ethola futhi ebona izimo zento entsha ngokufunda ukufunda into ibe yasendaweni kanye nokuhlukaniswa ngendlela ebumbene, ngokususa ukuqagela okuhlakaniphile kwesifunda

kanye nokubhala ngekhodi kokubili imifanekiso esekelayo nemibuzo yemifanekiso kube isigaba esithize. sezici ezibese zingena kudivayisi ejwayelekile yesigaba ukuze sikhiqize izibikezelo zezigaba ezithile. Siphinde sethula isifanekiso somphumela wokuhlunga inhloso evamile yemifanekiso ngokugcwele ngendlela yokuhlukanisa umfanekiso osetshenziselwa imisebenzi yokubona yekhompyutha eqoqweni lendlela yokufunda yomshini elihlanganisa isenzakalo ngasinye sibe uhlamvu oluthile futhi sikhiqize ukubikezela ngokuhlunga inhloso evamile yemifanekiso ngokuqondile, ngaleyo ndlela ibikezele kokubili izinto eziyisibonelo nezinto zasemuva ndawonye. Ngale ndlela, izakhiwo eziqaphelayo nezingaguquguquki ngokwezibalo zezenzakalo zento kanye nengemuva lazo zinganeliswa ngokulandelana kwazo ekuhambeni komsebenzi okuhlangene. Ekugcineni, sethula isifanekiso sezigaba ezimbili zolwazi oluhluziwe esenza isimo sokuphazamiseka sibe sikhulu sesici esishumekiwe semifanekiso kusetshenziswa ukulahlekelwa komsizi ozigadile. Izivivinyo kwamanye amaqoqo emininingwane endinganiso yendlela yokufunda yomshini ezinjengeminiImageNet, i-Omniglot, i-CIFAR-FS ne-Oxford Flowers102 yokuhlukaniswa kwemifanekiso, i-Pascal 5^i ne- COCO-20^i yokuthola into, kanye ne-Mapillary Vistas yokuhlukaniswa kwendlela yomfanekiso osetshenziselwa imisebenzi yokubona yekhompyutha ibonisa imingcele yokusebenza kanye nempumelelo yezindlela ezihlongozwayo zendlela yokufunda yomshini. Le ndaba ende ihlose ukuvala igebe phakathi kokufunda okujulile okujwayelekile nokufunda komuntu ngokudala izinhlelo zokubona zekhompyutha ezifunda ezibonelweni ezimbalwa zemininingwane yemifanekiso.

Amagama asemqoka:

Few-shot learning - Indlela yokufunda yomshini

Image classification - ukuhlukaniswa kwemifanekiso

Object detection - ukutholwa kwento

Knowledge distillation - ukuhluzwa kolwazi

Panoptic segmentation - indlela yokuhlukanisa umfanekiso osetshenziselwa imisebenzi yokubona yekhompyutha

Deep neural networks - ikilasi lokufunda ngomshini

Meta-learning - ukufunda ukufunda

Metric learning - ukufunda umsebenzi webanga phezu kwezinto

Image processing - inqubo yomfanekiso

Northern Sotho

Thaetlele ya thesese: Few-shot learning ya tlhopho ya diswantšho le temogo ya dilo

Thesese ye e laetša le go šomiša mekgwa e meswa e mmalwa yeo e šomanago le mathata a pono ya khomphutha a tlhopho ya diswantšho, temogo ya dilo, phetišo ya tsebo ya boitekolo le karogantšho ya dilo ka maemong a few-shot learning. Maikemišetšo a few-shot learning ke go tswalela sekgoba magareng ga mehuta ya go tsenelela ya go ithuta yeo e hwetšago tsebo go tšwa ditlhalošong tše di filwego tša dihlopha tša tshedimošo le batho ka mošomo o boima wa go ithuta ka mehlala ye mmalwa ye e hlalositšwego. Le ge dikgwekgwe tša pono ya dikhomphutha tše ntši di ka bonwa mošomong wo ka moka, se bohlokwa kudu ke mokgwa wo o lekaneditšwego wa datha wo o etšwego hloko. Re ela hloko maemo a few-shot learning moo mešongwana ya pono ya khomphutha ya go ba le datha ya thekgo ya yona le datha ya teko ya dipotšišo di amogetšwe gape ka ditiragalo. Re thoma ka go tsebagatša mmotlolo wa tlhopho ya few-shot learning le meta-learning tšeo di nago le baithuti ba bantši bao ba hlokomelwago ke molaodi wa bogare go laola go tlošwa ga dilo le meta-learning sephetho se se kopantšwego le kakaretšo. Sa bobedi, re tsebiša mokgwa wa temogo ya dilo wa few-shot wo o lemogang le go amogela mehlala ya dilo tše diswa ka meta-learning le tlhopho ka mokgwa wa botee, ka go tloša kakanyo ya kgakanego le go fetolela diswantšho tšeo di thekgang le diswantšho tša potšišo go dibopego tša karolo ye e ikgethileng go karolo ya tlhathollo ya go se kgodiše go tšweletša dikakanyo tša dikarolo tše itšego. Re tsebišitše gape mokgwa wo o feletšego wa go latela dilo tša go raragana ka go šomiša few-shot learning go romela dilo lefelong

le itšego gomme tša tšweletša kakanyo ya dilo thwii, gomme ya fa kakanyo ya ditiragalo tša dilo tše pedi mmogo. Ka tsela ye, mehlala ya mokgwa wo le dilo tšeo di latelanago tša semanthiki tša mešomo ya dilo le botšo bja tšona di ka latelana gabotse go mešomo ye e kopantšwego. Mafelelong re tsebiša mmotlolo wa phetišo ya tsebo wa dikgato tše pedi woo o kaonafatšago maemo a go raragana ga dilo tše di lokelwago diswantšhong ka go šomiša tahlegelo ya thekgo ya boihlokomelo. Boitekelo godimo ga dihlopha tša datha tše dingwe tša bohle tša tekanetšo ya few-shot learning bjale ka miniImageNet, Omniglot, CIFAR-FS le Oxford Flowers102, Pascal 5^i le COCO-20^i ya temogo ya dilo, le Mapillary Vistas ya karogantšho ya dilo go laetša magomo a tshepedišo le go šoma gabotse ga mekgwa ye e šišintšwego ya few-shot learning. Maikemišetšo a thesese ye ke go tswalela sekgoba magareng ga thuto ye e tseneletšego ya tlwaelo le go ithuta ga batho ka go hlama mananeo a pono a khomphutha ao a ithutago go tšwa mehlaleng ye mmalwa ya datha ya diswantšho.

Mantšu a bohlokwa: few-shot learning (FSL), tlhopho ya diswantšho, temogo ya dilo, phetišo ya tsebo, karogantšho ya dilo, dineteweke tša nyurale ye e tseneletšego (deep neutral networks (DNN)), meta-learning, thuto ya metriki, peakanyo ya diswantšho

Nomenclature

2D Two-Dimensional

3D Three-Dimensional

ANN Artificial Neural Networks

CNN Convolutional Neural Networks

DCN Deformable convolutional networks

DL Deep Learning

DETR DEtection TRansformer

DNN Deep Neural Networks

FSL Few-shot Learning

FC Fully Connected

FCN Fully Connected Neural Networks

FPN Feature Pyramid Network

FSL Few-shot Learning

FSOD Few-shot Object Detection

GANs Generative Adversarial Networks

GPU Graphics Processing Unit

JSON Javascript Object Notation

LSTM Long Short-Term Memory

LSTD Low-Shot Transfer Detector

MAML Model-agnostic meta-learning

mAP mean Average Precision

MAP Masked Average Pooling

MPSR Multi-scale Positive Sample Refinement

MS COCO Microsoft Common Objects in Context

NMS Non-maximum suppression

ONCE OpeN-ended Centre nEt

PQ Panoptic Quality

ReLU Rectified Linear Unit

RNNs Recurrent Neural Networks

R-CNN Region-Based Convolutional Neural Networks

RoI Region of Interest

RPN Region Proposal Network

SAM Semantic Alignment Mechanism

SGD Stochastic Gradient Descent

SSD Single-Shot Detection

TPU Tensor Processing Unit

VAE Variational AutoEncoder

ViT Vision Transformer

YOLO You Only Look Once

YOLOR You only Learn One Representation

Definition of Terms

Attention: A mechanism that equips a neural network with the ability to focus on a subset of important input features, and devote more computing power to that small but important part of the input, described as soft attention or hard attention, and can be global, or local. For matrix-valued inputs such as images, it is referred to as visual attention, and it is implemented as element-wise multiplication.

Base set: Feature maps that have been acquired through transfer learning from a task with a large amount of data to learn a representation of some inputs for further comparison in the feature space. The base set can then be used with a new model where data is not abundantly available.

Bipartite matching: An algorithm mainly used with transformer networks that finds the best match between the ground truth and the predictions, which minimises the error between them.

Contrastive loss: A metric learning loss which operates on two given data points (X_1 and X_2) produced by a network, e.g. by a Siamese network, and their positions relative to each other, and specifies whether they are similar or dissimilar.

Convolution: A mathematical operation on two functions ($\mathbf{f}$ and $\mathbf{g}$) that produces a third function $(f \star g)$ that expresses how the shape of one is modified by the other. The CNN repeats the application of the same filter to an input that results in a map of activations also known as a feature map for an image, indicating the strength and the locations of a detected feature in an input space.

Convolution filter: Is a matrix of weights comprised of integers that is applied to an image with mathematical operation. It works by determining the value of a central pixel by adding the weighted values of all its neighbours together. In CNNs, the value of each filter is learned during the training process. It is also known as a **kernel**.

Convolutional neural network (CNN): A feed-forward neural network that is generally used to analyse visual images by processing data with grid-like topology. It contains many convolutional layers stacked on top of each other, each one capable of picking up patterns in the input image such as lines, gradients, circles, or even eyes and faces, and therefore used to classify images, or detect and classify objects in an image.

Cosine similarity: A measurement that quantifies the similarity between two or more vectors, described mathematically as the division between the dot product of vectors and the product of the Euclidean norms or magnitude of each vector.

Cross entropy loss: A classification loss which operates on class probabilities produced by the network independently for each sample, used when adjusting model weights during model training.

Data augmentation: A technique used to artificially create new training data from existing training data by putting in slightly modified versions of already existing data into the existing dataset, or adding newly created synthetic data from existing data with the intention of expanding the training dataset with new, plausible training examples. It acts as a regulariser and helps reduce over-fitting when training a machine learning model.

Deep learning: Subfield of machine learning based on artificial neural networks, which can be thought of as learn hierarchical representations of the input data through non-linear transformations, with architectures that include CNNs, RNNs, and LSTM, among others.

Embedding: A relatively low-dimensional, learned continuous vector representation of discrete variables useful for reducing the dimensionality of categorical variables and meaningfully representing categories in the transformed space. It captures some of the semantics of the input by placing semantically similar inputs close together in the embed-

ding space, making models more efficient and easier to work with.

Encoder-decoder architecture: Deep learning architecture that can handle inputs and outputs of variable-length sequences suitable for sequence transduction problems where an input sequence is read in entirety and encoded to a fixed-length internal representation. An **encoder** first takes a sequence of variable-length as input and transforms it into a state with a fixed shape. A **decoder** maps the encoded state of the fixed shape to a sequence of variable-length.

Feature extraction: A machine learning process for methods that select and/or combine variables into features, whereby the initial set of raw data is reduced to more manageable groups for processing, effectively reducing the amount of data that must be processed. The product will still be able to accurately and completely describe the original dataset.

Feature Pyramid Network (FPN): A top-down architecture with lateral connections used as a generic feature extractor for building high-level semantic feature maps of an input image at all scales that has been used with object detection systems. The feature extractor takes a single-scale image of an arbitrary size as its input, and the resulting output has proportionally sized feature maps at multiple scales and levels.

Few-shot learning: A type of machine learning problem that uses CNNs (specified by experience **E**, task **T** and performance **P**), where **E** contains only a limited number of examples with supervised information for the target output.

Generative Adversarial Networks (GANs): A deep learning model based on the idea of adversarial training that consists of two neural networks that compete against each other. The **generator** is a convolutional neural network that artificially manufacture outputs that could easily be mistaken for real data, and the **discriminator** is a deconvolutional neural network that identifies which outputs it receives have been artificially created.

Hyperparameter: A neural network variable that is tuned before training the model like number of layers, activation functions, loss functions, optimizers, early stopping, learning rate.

Knowledge distillation: A procedure for model compression, whereby to train the small deep learning model called the student, on a transfer set with soft targets provided by the large model also known as the teacher, usually by compressing the knowledge of a large and computational expensive model (often an ensemble of neural networks) to a single computational efficient neural network.

L1 and L2 Regularisation: Regularisation is a technique to penalise complex deep learning models to reduce over-fitting by making network weights small. L1 regularisation adds a penalty that is equal to the absolute value of the magnitude of coefficient, or restricts the size of the coefficients. L2 regularisation adds a penalty equal to the square of the magnitude of coefficients.

Logits layer: The raw scores, or prediction values that are produced as real numbers ranging from [-infinity, +infinity] that are output by the last layer of a neural network and are fed into the Softmax layer which turns them into probabilities and used for a classification task in neural networks.

Loss function: A measure used to determine the error or the loss between the prediction of the neural network with respect to the expected output label or given target value.

Mask classification: Image segmentation task whereby the image is partitioned into N regions represented with binary masks, associating each region as a whole with some distribution over K categories, as opposed to per-pixel classification whereby the individual image pixels are analysed by the spectral information they contain.

Mask transformer: Attention-based neural network, which consists of two sublayers, namely, Self-Attention Network (SAN) and Feed-Forward Network (FFN). It basically transforms a given sequence of elements, such as the sequence of words in a sentence, or image patches into another sequence.

Mean squared error: The mean of the squared prediction errors over all instances in the test set. The prediction error is the difference between the true value and the predicted value for an instance.

Meta-learning: An approach to machine learning whereby models are designed to learn new skills or adapt to new environments rapidly across a suite of related prediction tasks with a few training examples. Also referred to as "Learning-to-learn".

Metric-learning: An approach in deep learning based directly on a distance metric that aims to establish similarity or dissimilarity between objects. It aims to reduce the distance between similar objects, and to increase the distance between dissimilar objects.

Neural style transfer: The technique of transferring the style such as texture, colours, and other visual patterns from one image to another.

Non-maximum suppression (NMS): A class of algorithms used mainly in object detection to select one best bounding box out of many overlapping boxes, using, for instance, some form of probability number and some form of overlap measure (e.g. IoU) for further processing.

Object detection: A deep learning process that locates the presence of objects with a bounding box or mask, and types or classes of the located objects in the input image. The output is one or more bounding boxes usually defined by a point, width, and height, or a mask that covers the identified image, and a class label for each bounding box or mask.

Object localisation: To predict the presence of a bounding box or mask around the object if present in the image.

Optimisation: The problem of finding a set of inputs to an objective function that results in a maximum or minimum function evaluation. Gradient Descent, Adaptive Learning Rate, and Stochastic gradient descent (SGD) are some of the most used optimization algorithms for deep learning.

Over-fitting: An analysis by a neural network model that corresponds too closely or exactly to a particular set of data, and therefore fails to generalise and predict future observations reliably. This is the main cause of poor performance in deep learning. Approaches to reducing overfitting include L1 regularization, adding dropout layers, and artificially increasing the size of the training set.

Panoptic segmentation: A type of segmentation that unifies the two distinct tasks of semantic segmentation that assigns a class label to each pixel of an image and instance segmentation that detects and segments each object instance in an image.

Parameter: A variable that is updated by the network during the training process such as weights and biases that are tuned automatically by the model during training, as opposed to hyperparameters that are set by the user before training.

Self-supervised learning: A machine learning approach that does not depend on the humans to label and categorise the training objects. The features, and the labels are learned first from unlabelled data in what is called representation learning. The real model is then learnt from the features extracted from the labelled data. The labels are generated from the given data.

Semantic Alignment Mechanism (SAM): A mechanism used to facilitate meta-learning with deep learning networks that orient high-level image feature semantics to low-level feature semantics of images to improve model generalisation capabilities of meta-learnt representations.

Siamese network An artificial neural network that uses the same weights while working in tandem on two different input vector representations to compute comparable output vectors.

Stochastic gradient descent (SGD): An iterative optimization algorithm for optimizing an objective function that randomly picks one instance in the training set for each one step and calculates the derivative of gradient based only on that single instance and calculating the update immediately. SGD is used to find the optimal parameter configuration for a deep learning algorithm by iteratively making small adjustments to the network configuration to decrease the error of the network.

Support set and query set: In few-shot learning, we have n-labelled examples of each K classes, that is $N \times K$ total examples which we call support set S. We also have to classify a query set Q, where each example lies in one of the K classes.

Transfer learning: A technique whereby an existing model trained on one dataset usually with a high level of generalisation is reused to solve a problem in another domain by fine-tuning the former network model, keeping the weight fixed, or adapting them entirely when training the model to save both training resources and time.

Triplet loss: A loss function defined on triples of images where a baseline (anchor) image input is compared to a positive input and a negative input. The function makes the distance of the embeddings between the anchor and a positive image, to be a positive, i.e. the images are of the same type, whereas, in contrast, the anchor image when compared to the negative example results will result in them having a larger distance between them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

In this chapter, the general context of this thesis is introduced, including the introduction to few-shot learning, the aims and objectives of the study, the major results, and the thesis contributions. The outline of the thesis is provided.

1.2 Supervised and Unsupervised Learning in Deep Learning

Over the past decade, there has been heightened interest in the use of deep neural networks (DNNs) in computer vision applications. The goal is for algorithms to learn common patterns from vast amounts of image data often with millions of images that have been hand-annotated. For example, the ImageNet [43] is a large dataset of more than fourteen (14) million visual images of approximately 150GB size that has been widely used in visual object recognition research. The other datasets that have also been used in image recognition are the MS-COCO [140] and Mapillary Vistas [161], among many others. A common scenario for these datasets is one in which labels are costly and time-

consuming to acquire from abundant data, for instance, collecting video, audio and image data is cheap with today's cameras, but high-precision labelling of the data is costly and cumbersome, making the whole process infeasible. Deep learning networks are, however, incapable of adapting to new unseen data or environments. Whenever new data is available, the deep neural network has to be re-trained to incorporate the new patterns to be able to generalize, which becomes infeasible in the current world where new data becomes available every time.

Deep learning refers to a broader family of machine learning methods based on artificial neural networks (ANNs) with representation learning. These learning methods can generally be categorised into supervised, semi-supervised or unsupervised. With supervised learning, the training set is submitted as input to the system during the training phase. Each input is labelled with a desired output value. In this way the model is provided with both the input and the corresponding output. Models need to find the mapping function to map the input variable (X) with the output variable (Y). In unsupervised learning, the model looks for previously undetected patterns in a dataset with no pre-existing labels and with a minimum of human supervision. Semi-supervised approaches combine a small number, depending on the size of the dataset, of labelled data with a large amount of unlabelled data during training. It falls between supervised learning and supervised learning.

Deep learning models [117, 124, 212, 225] have become exceptionally good at learning functions that map inputs to human-generated labels under one condition that an enormous amount of labelled data must be fed to them first. There is also another drawback, that these models, determined to classify the input into a category, do not learn much about the inherent properties of input elements. The feedback the machine is given is scarce in supervised learning, so naturally the networks are very sample inefficient. This creates a significant issue that high-quality data is often hard to come by for a lot of applications and obtaining an annotated dataset can prove too costly an undertaking even for

large organizations.

In computer vision tasks, deep learning has been successfully managed by convolutional neural network (CNN) learning models. These include models for object recognition and detection, image classification, person identification, activity recognition, among others, tasks that are useful in many fields of application, including autonomous driving, medical image analysis, robotics and video surveillance. The drawback of these deep learning approaches is that they require a huge amount of annotated training data for supervised training, and computing resources for these tasks are generally expensive or unavailable. Data annotation is usually an expensive and tedious task. On the other hand, data can also be rare or difficult to gather for some reasons, including safety and ethical issues. Moreover, a deep learning model trained successfully for a specific task A, cannot be directly deployed for task B, even though they are similar. This means that a learning algorithm can only be good at mastering one task. On the other hand, humans learn by combining and generalizing multiple concepts. It is therefore essential to develop models that can learn from few annotated samples of training data. Few-shot learning, a subfield of machine learning aims to narrow the gap between machine and humans in the challenging task of learning from few examples. The aim of this thesis is to propose novel methods that optimize the model ability to detect and recognise new object instances using few labelled data. The idea is to bridge the divide between conventional deep learning and human learning. This can be achieved by creating systems that learn from a few examples of data.

1.3 Few-shot Learning

In few-shot learning, the terms support set and query set are used to describe the dataset for training and for testing. In traditional supervised learning, the test set has not been seen before, though its examples belong to one of the classes that have been used during

training. Test samples are from known classes. For instance, if the model is being used to classify dogs and cats, only samples of dogs and cats are used for training. Sheep cannot be part of either the training set or the test set in traditional supervised settings. Few-shot learning is a different problem. The query has not been seen before. It is from an unknown class. The support set could have samples of dogs, cats, and elephants. The few-shot model could have a sheep as part of the query set. The training set does not have the sheep class.

The idea behind few-shot learning is to learn a similarity function $sim(X, X')$ which measures the similarity between X and X'. The ideal situation is that $sim(X1, X2) = 1$, $sim(X1, X3) = 0$, and $sim(X2, X3) = 0$ (refer to Figure 1.1.

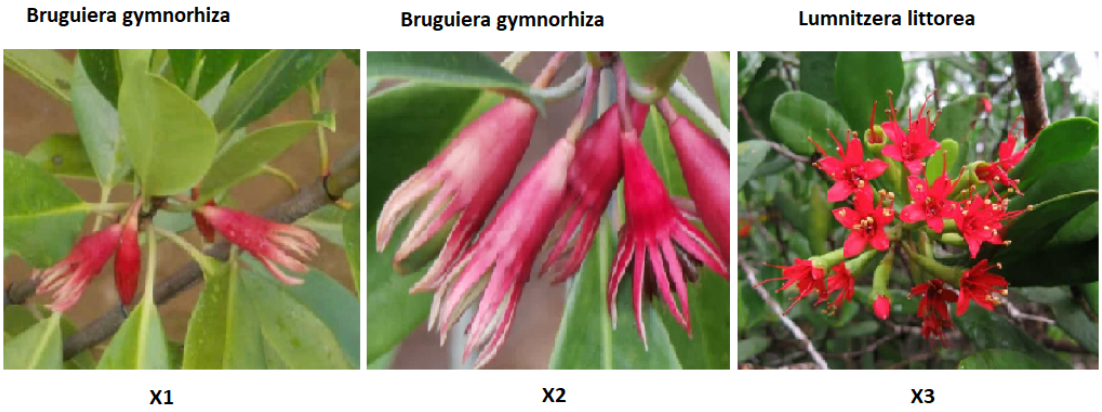


Figure 1.1: Illustration of the similarity between the support and the query sets for mangrove species.

The basic idea is to first train a model using a large training dataset that has been successfully used for classification such as the ImageNet dataset. The model can then be quickly trained for novel datasets to adapt it. For instance, a model trained on a vast dataset of images that includes vegetation types, animals, cars, and buildings can quickly adapt to the training for the identification of tree species.

1.4 Task Formulation

Few-shot learning aims to produce a model such that, given a learning episode with N classes and a few labelled examples k_c per class $c \in 1, \ldots, N$, the model is able to generalise to novel labels for that episode. The new model has learnt from a support set $S = \{(x_1, y_1), \ldots, (x_K, y_K)\}$, $K = \sum_c K_c$. The model is assessed and evaluated on a query set $Q = \{(x_1^*, y_1^*), \ldots, (x_T^*, y_T^*)\}$. x, y is the input image vector $x \in \mathbb{R}$ and the image label $y \in 1, \ldots, N$. Episodes are described as $N - way\ K - shot$ episodes. Training often proceeds in an episodic fashion.

1.5 Relation to Other Common Machine Learning Sub-Fields

There are some machine learning subfields that are related to few-shot learning. We will describe continual learning, transfer learning, and open set recognition briefly and highlight the main differences with few-shot learning.

Continual learning. Continual learning, also known as sequential learning, incremental learning, or lifelong learning is also a type of machine learning that studies the problem of learning from an endless or infinite stream of data stemming from changing input domains. It is associated with different learning tasks and problem domains, with the goal of using the acquired knowledge in problem solving and future learning [3, 156, 175, 271]. Continual learning models continuously learn based on the input of increasing amounts of data while using previously acquired knowledge. It is motivated by the Stability–Plasticity Dilemma [4] in humans who have astonishing ability to adapt by effectively acquiring knowledge and skills, refining them on the basis of novel experiences, and transferring them across multiple domains. The algorithms are designed to accumulate and improve knowledge in a curriculum of learning-experiences, without suf-

fering from catastrophic forgetting experienced by deep learning systems when exposed to new sets of datasets. For most applications, it is infeasible to annotate all training labels from all envisaged tasks before the start of the model learning process. Therefore, continual learning constantly evolves and should adapt to the changing world, and models must adapt and continue to learn. All previously seen data should have their dimension reduced for such a system or process to be efficient. There is no need for a large scale full re-training at each point. Continual learning is a more general formulation of few-shot learning.

Transfer Learning. Transfer learning [272] aims at "aiding the learning process of a given task by exploiting the knowledge of another model" [272, p. 1] such as InceptionV3 [226], or ResNet [83], and others, that has been trained using a different dataset domain such as Imagenet or Microsoft COCO, to save time and resources, and also to train a model where there is a shortage of training data. Deep neural networks are immensely data-hungry and rely on huge amounts of labelled data to achieve high performance. It has been observed that a deep neural network trained in computer vision images or video shows that the first layers of the network learn the general features like when using Gabor filters. The layers become steadily more particular to the information about the class types that are present in the input image. Deep neural networks can take several days of training, so the weights that have been learnt on a larger model can be used and fine-tuned to save training time. Domain adaptation is a sub-field of transfer learning which deals with situations in which a model trained on one image dataset distribution is thereafter used in the context of a different dataset target. It may also use labelled data in one or more source domains to solve novel tasks in another target domain where the source and target tasks are the same but drawn from different input domains. The target dataset is usually unlabelled. The primary objective is to adapt a model trained on the source domain to perform well on the unlabelled target domain without changing how the model works. Some successful few-shot learning models derive feature representations through

transfer learning.

Open Set Recognition. Open set recognition aims to classify the known and recognize the unknown where there exists partial data from the dataset at training time. The models should be able to predict known and unknown classes submitted for classification during testing, requiring the classifiers to not only accurately classify the seen classes, but also effectively deal with unseen ones. The set is "open" because we want to classify what we can amongst the closed set of classes that we have, but we want to classify samples from the open world, therefore open set. It can be categorized into four classes, 1) known knowns, that is, labelled images of classes which we want to recognize; 2) known unknowns - unlabelled images that do not belong to any of the classes that we want to recognize, 3) unknown known classes - classes that we have no samples of but know that exist through side information; and 4) unknown unknown class - classes that we have no samples of and we do not know that exist.

1.6 Aim

The goal of this thesis is to study few-shot learning. This thesis presents several novel methods that tackle the problems of image classification, object detection, self-supervised knowledge distillation, and panoptic image segmentation in few-shot learning settings.

1.7 Contributions

Even though multiple computer vision themes can be identified throughout this work, the most important is the limited data regime taken into account. We consider the few-shot learning setting where tasks associated with their support and query test data are received and trained in episodes.

Before presenting the main contributions, we review some of the most relevant literature in Chapter 2. The purpose is to give a general overview of the literature related to

few-shot learning (Section 2.2) including approaches to few-shot learning (Section 2.3), object detection (Section 2.5), knowledge distillation (Section 2.6), panoptic segmentation (Section 2.7.2), the Vision Transformer (Section 2.8), and commonly used datasets for few-shot learning (Section 2.9). Chapters 3 to Chapter 6 contain separate related work sections, more specific to their very content.

In Chapter 3, we introduce a novel meta-learning model that consists of dual learners composed of a pre-trained encoder, and supervised by a central controller to control modules for feature extraction and meta-learning, and a meta-ensemble module for integrated inference and generalisation. In particular, each meta-learner is fine-tuned by batch training and parameter-free decoder used for prediction. First, ResNet152 is used as a backbone to learn a representation f_θ of input on base set. We then optimize the classifier by using the cosine distance in the feature space in the meta-training stage. We provide some insights for best practices in implementation on the Omniglot, miniImageNet and Oxford Flowers 102 datasets.

Chapter 4 introduces a novel approach to fully convolutional few-shot multi-scale object detection in input images. Our approach meta-learns object localisation and classification in a unified manner by eliminating region-wise prediction. Both support images and query images are encoded into category-specific features first. They then feed into a category-agnostic decoder to generate predictions for the specific image categories. To facilitate meta-learning, a module designed in multi-scale architecture to enable multi-scale object detection is designed. This model aligns semantics of high-level feature and low-level feature representations. Experiments on two public benchmark datasets, Pascal 5^i and COCO-20^i demonstrate the proposed method's effectiveness for "few-shot object detection" [55, p. 1].

Chapter 5 introduces a novel fully convolutional model for few-shot learning for panoptic segmentation. The model encodes each instance image into a specific kernel and generates a prediction by convolutions directly, thereby predicting both instance im-

age objects and background regions together. In this way, instance-aware and semantically consistent properties for things and stuff can be respectively satisfied in a unified workflow. Experiments on the Mapillary Vistas dataset demonstrate the effectiveness of the proposed method for few-shot panoptic segmentation.

In Chapter 6, we introduce a two-stage deep learning knowledge distillation model that uses a self-supervised loss to create augmented images. The model maximises the entropy of the feature embeddings to estimate an optimal output manifold in the first stage. The model reduces the gap between the embeddings while fixing the original image samples to the learned embedding manifold using a distillation loss for few-shot classification of the input images. The entropy is minimised on feature representation by bringing self-supervised twins together. It simultaneously constrains the manifold with student-teacher knowledge distillation.

Finally, we conclude in Chapter 7 by outlining the main findings of the thesis and by suggesting potential avenues for future research.

1.8 Outline of the Thesis

The rest of the manuscript is organized as follows: We present and discuss from a broad perspective works on few-shot learning that emerged during the past few years in Chapter 2. Our novel work on few-shot image classification is detailed in Chapter 3. We also present a novel approach for "few-shot object detection" [101, p. 1] for learning object localisation and categorisation in Chapter 4. A fully convolutional model for few-shot panoptic segmentation is proposed in Chapter 5. We move to present our few-shot two-stage knowledge distillation model that maximises the entropy of feature embeddings using a self-supervised auxiliary loss in Chapter 6. We outline the main findings, and future directions in Chapter 7.

1.9 Conclusion

This chapter provides the general context of the thesis including the introduction to few-shot learning in deep learning settings, the aims and objectives of the study, and the thesis contributions. The following chapter provides a comprehensive review of some of the most relevant literature in few-shot learning, including a general overview of the literature related to few-shot learning in image classification, object detection, knowledge distillation, the Vision Transformer, and panoptic segmentation.

Chapter 2

Literature Review

2.1 Introduction

We are interested in studying the problem of few-shot learning in deep learning with computer vision. In particular, our focus is on improving methods for image classification, object detection, panoptic segmentation, and knowledge distillation in few-shot learning settings, where models are trained with a small amount of data in episodic fashion using support set images and corresponding query images. Most benchmarks combine various methods, including data augmentation methods, metric-based methods, models-based methods and optimisation-based methods in all areas of computer vision. Few-shot learning presents several challenges in computer vision, including:

- Improving feature representation in classification, object detection and segmentation,
- Identifying object instances of interest given only a small amount of training data,
- Differentiating individual object instances,
- Simultaneously recognizing both semantic labels and object instances, and

- Designing end-to-end models for few-shot learning.

Therefore, this thesis presents several novel methods that tackle the problems of image classification, object detection, self-supervised knowledge distillation, and panoptic segmentation in few-shot learning settings. We consider the few-shot learning setting where tasks associated with their support and query test data are received and trained in episodes.

This chapter explores the literature pertaining to all relevant research on the methods and techniques in few-shot learning for image classification, object detection, knowledge distillation and panoptic segmentation. Current research topics that have addressed the problem of learning with limited data or few-shot learning in deep learning with image processing, including seminal work in meta-learning, metric learning, data augmentation, knowledge distillation, the vision transformer and self-supervised learning are explored. We give a brief review of literature for few-shot learning and few-shot image classification in Section 2.2. We explain approaches that have been used for few-shot learning in Section 2.3. A brief overview for few-shot learning on object detection follows in Section 2.5. We then present a knowledge distillation view on few-shot learning in Section 2.6. We give an overview of panoptic segmentation in Section 2.7, including semantic and instance segmentation. Finally, we describe few-shot seminal work on the Vision Transformer (ViT) in deep learning in Section 2.8. The focus is on few-shot learning in image processing.

2.2 Few-shot Learning

After the success of deep learning thanks to the powerful Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs) allowing training on large datasets and deep architectures, few-shot learning has received increased attention in recent years. Few-shot learning [56, 57, 121, 239], or the ability to learn from few labelled samples has

been motivated by human-centred intelligence that has the ability to learn novel objects on the fly from few samples. The current successful deep learning models for classification, image segmentation, and object detection and recognition are based on supervised learning. They train with large-scale datasets such as ImageNet, Microsoft COCO and Open Images with millions of labelled samples. However, even with large-scale datasets they remain limited in multiple aspects. Not all objects in our lives are within the one thousand (1000) labels provided per object type, for instance on ImageNet. There is therefore, need to develop models that can learn from few samples of data. Training an accurate deep learning model using only a few training examples is a particularly challenging problem. Successful models learn patterns very slowly, and require vast amounts of processing capabilities. The main goal of developing few-shot learning models and techniques is to develop deep learning models that are able to infer from specific cases better to novel classes given a quantity of samples. This is normally achieved with iterative training based on the stored knowledge representations or embeddings acquired from training the labels on a large annotated dataset.

Few-shot learning aims to build deep learning models that can learn efficiently to recognize patterns in the low data regime, applied in situations where image data is scarce. Early works [56, 58] approaching few-shot learning focused on one-shot learning where the distance function, or regularisation terms such as $L1$, $L2$; or the probability models were used to classify objects. Several directions have since been explored such as metric learning [88, 103, 104, 113, 224], data augmentations [2, 22, 68, 253, 272], meta-learning [61, 239], memory augmented networks [206], or combined model architectures [88, 217]. More recently, the most popular and effective learners have addressed few-shot learning using optimisation-based models, e.g. meta-learning and/or metric-learning-based methods. The key idea is to leverage a large number of similar few-shot tasks in order to learn how to adapt a base-learner network to a new task for which only a few labelled examples or samples are available, and learn only features that are distinctive or

similar in the images. The critical challenges that have been encountered are fast adaptation without over-fitting, time and resource efficiency, and generalisation across different datasets.

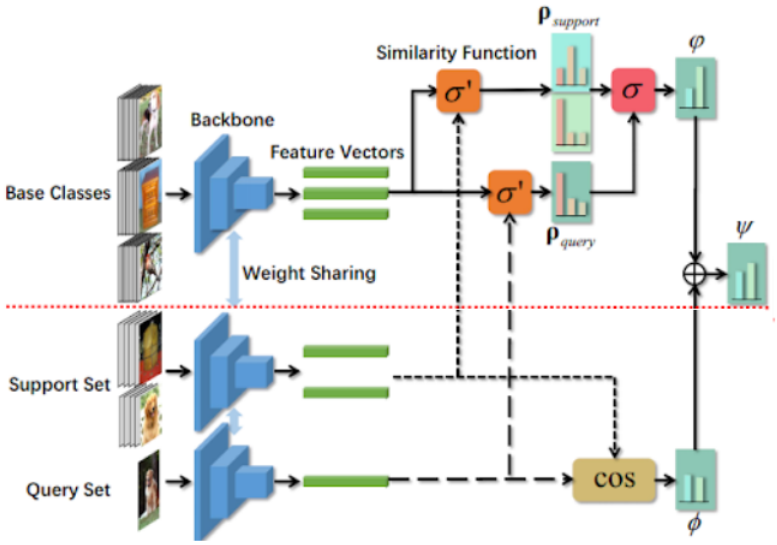


Figure 2.1: An illustration of few-shot learning using metric learning.

Various approaches have been employed for few-shot learning. Gradient descent-based approaches [5, 61, 190] learn to rapidly adapt a model to a task via a small number of iterations during processing. Metric learning-based [113, 214, 222, 239] (e.g. the Euclidean Distance, or the Minkowski Distance) approaches learn a distance metric between a query image, and the support images. Other approaches learn to map a test example to a class label by accessing memory modules that store training examples for that task [206]; and others learn how to generate the weights of a classifier [71, 182, 273], and methods that "hallucinate" additional examples of a class from a reduced amount of data [82, 151, 279]. Few-shot learning has also been employed in self-supervised learning [47, 71, 199, 227, 288] where it focuses instead on unlabelled data and looks into it for the supervisory signal to feed deep neural networks.

Many circumstances exist where accruing enough data to increase the accuracy of

deep learning models is unrealistic. Few-shot learning models are driven by the concept that reliable algorithms can be created from minimalist datasets. Some of the driving factors behind its increasing interest and adoption include the reduction of data collection and computational costs since few-shot learning requires less data to train a model. Additionally, in the event of scarcity of data or rare-case learning, supervised machine learning techniques find it challenging to make accurate predictions and make accurate inferences.

Definition Consider a machine learning task **T** (see Table 2.1). A machine learning program can improve its performance **P**, e.g. for classification, object detection, or image segmentation through experience **E** obtained by training on a large number of labelled images. The program improves its classification accuracy **P** by training on a database of experiences **E** of millions of images annotated by human experts. **Few-shot Learning** is a type of machine learning problem specified by **E**, **T** and **P**, where **E** contains only a limited number of examples with supervised information for the target [242, p. 1.5].

Table 2.1: Machine learning task

Task **T**	Experience **E**	Performance **P**
Image classification	Large-scale labelled dataset	Classification accuracy

Many successful deep learning approaches employ supervised learning models, and these require that training samples be labelled. However, sometimes accruing enough data to increase the accuracy of the model is unrealistic and difficult to achieve. Labelling samples is generally costly. For instance, labelling x-ray images requires the involvement of medical specialists. Few-shot learning may also be of great assistance to discover patterns in confidential medical data or rare plant or animal diseases, and make beneficial predictions. Humans are known to learn from very few examples. Unlike deep learning models, humans do not need thousands of dog and cat images to distinguish between the two animals. Even a child can learn what a dog is after seeing one or two examples of

these animals. Few-shot learning aims to close this gap between deep learning models and humans in the challenging task of learning from few examples. There is need, therefore, to develop and improve ways to learn from few examples in deep learning.

A few-shot learning model is fed with a very small amount of training data, ideally between 1 and 5, and sometimes up to 20 training examples and corresponding labels, contrary to the deep learning practice of using a large amount of annotated data. For instance, if we have a problem of categorizing tree species from aerial photos, some rare species of trees may lack enough pictures to be used in the training images. If we have only one image of a certain tree, this would be a one-shot machine learning problem. In extreme cases, where we do not have every class label in the training, and we end up with zero training samples in some categories, it would be a zero-shot [9] machine learning problem.

2.2.1 Few-shot learning terminology

In few-shot learning, the terms **support set** $(\mathcal{S})$ and **query set** $(\mathcal{Q})$ are used to describe the dataset for training and testing. For instance, a few examples are sampled from each class from the dataset $\mathcal{D}$ and assigned as support set. Similarly, some other corresponding data points are sampled from each class and assigned as the query set. In this setup, the model is trained on the **support set** and tested on the **query set**. Moreover, the few-shot learning models are trained in an episodic manner, such that, at each episode, new data points are sampled from the dataset $\mathcal{D}$ and assigned as support and corresponding query set. This means that at each episode the model is trained and tested on different support and query sets. After a series of episodes the model has learned how to learn from a smaller dataset [239].

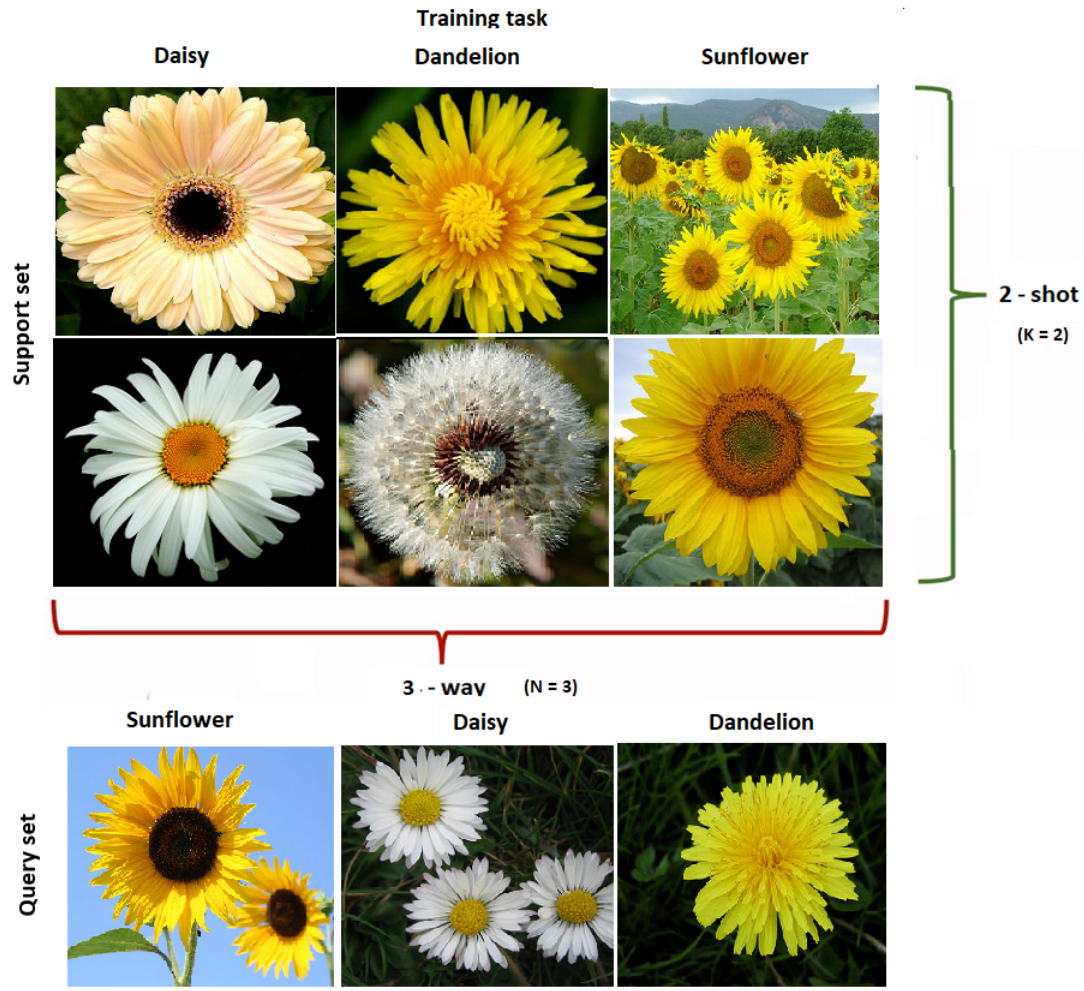


Figure 2.2: Few-shot learning: The model is trained using a series of similar training tasks. The task in this instance is a 3-way-2-shot classification problem, which, in this case, each training task contains a support set with three different classes of flowers; daisy, dandelion and sunflower, and two examples of each flower species. During training the objective or loss function is used to assess the performance on the image query set for each task in turn given the respective support set images. At test time, a completely different set of tasks is used. The query set is used to evaluate the performance, given the support set. There is no overlap between the classes in the individual training tasks and between those in the test task. The model or algorithm must learn to classify image classes in general rather than any particular set.

The terminology commonly used in few-shot learning to describe the problem setup is **N**-way **K**-shot learning (see Figure 2.2). **N**-way **K**-shot learning refers to the number of classes and the number of data samples from each class available during training to be used as support set and query set. The aim is to discriminate between **N** classes with **K** examples of each class. A typical problem size might be to discriminate between **N=10** classes with only **K=5** data samples from each class to train from. With this amount of data samples, we cannot train a classifier using conventional deep learning methods that depend on far more parameters than there are training examples. It will not be able to generalize.

For example, if the task is to classify mangrove trees and non-mangrove trees, then it is a 2-way setup as there are only 2 classes (mangroves and non-mangroves) to be learned. Moreover, during training, if only 5 samples of mangroves trees and 5 samples of non-mangroves are introduced to the model as support set, then it is a 2-way 5-shot learning setup. The goal of few-shot learning is to classify new image data having seen only a few training examples. In practice, few-shot learning is useful when training examples are hard to find and rare for using normal deep learning models, or where the cost of labelling data is high, such as diagnosis of rare diseases. If the data is insufficient to constrain the problem, then one possible solution is to gain experience from other similar problems from the same domain.

The capability and the performance of an algorithm to perform few-shot learning is typically measured by its performance on the **N**-way, **K**-shot tasks. These tasks are run as follows:

- A model is given a query sample belonging to a new, previously unseen class.
- It is also given a support set, $\mathcal{S}$, consisting of **N** examples each from **K** different unseen classes.
- The algorithm then has to determine which of the support set classes the query

sample belongs to.

2.2.2 Few-shot learning notation

The primary goal of few-shot learning is to make a model generalise its performance to novel categories from a handful of samples with iterative training based on prior knowledge acquired from training samples on a labelled dataset consisting of a large number of samples.

Table 2.2: Few-shot learning notation.

Commonly used Notation in Few-shot Learning	
Symbol	Description
$\mathcal{D}_{train}$	Training set
$\mathcal{D}_{test}$	Testing set
(x_n, y_n)	n number of samples and their labels in $\mathcal{D}_{train}$
(x_n^i, y_n^i)	n number of samples and their labels in $\mathcal{D}_{test}$
h	Hypothesis
$\hat{h}$	Hypothesis for meta-learning
θ	Model parameters
(S)	labelled Support set
(Q)	Query set
$\mathcal{T}_i$	i set of tasks where each task is a set of classes

Consider a learning task $\mathcal{T}$ (see Table 2.2), few-shot learning (FSL) deals with a dataset $\mathcal{D} = \{(\mathcal{D}_{train}, D_{test})\}$ consisting of a training set $D_{train} = (x_i, y_i)_{i=1}^{I}$ where I is small, and a testing set $D_{test} = \{x^{test}\}$. Let $p(x, y)$ be the ground-truth joint probability distribution of input x and output y, and $\hat{h}$ be the optimal hypothesis from x to y. FSL learns to discover $\hat{h}$ by fitting D_{train} and testing on D_{test}. To approximate $\hat{h}$, the

FSL model determines a hypothesis space $\mathcal{H}$ of hypotheses $h(.;\theta)$'s, where θ denotes all the parameters used by h. Here, a parametric h is used, as a non-parametric model often requires large datasets, and thus not suitable for FSL. A FSL algorithm is an optimization strategy that searches $\mathcal{H}$ in order to find the θ that parametrises the best $h^\star \in \mathcal{H}$. The FSL performance is measured by a loss function $\ell(\hat{y}), y$ defined over the prediction $\hat{y} = h(x;\theta)$ and the observed output y.

2.3 Few-Shot Learning Approaches

Few-shot learning methods have adopted an episode-based training strategy to learn meta-knowledge that enables the model to adapt to new tasks that contain unseen classes with only a few samples. Common approaches (refer to Figure 2.3) that have been used are generative and data augmentation based (also known as hallucination methods), metrics-based learning, also known as embedding-based (learning a general metric), optimization-based (learning to optimize the model quickly), and models-based methods ((learning to accumulate and generalize experience) [96]. An approach by [242] categorised existing supervised few-shot learning methods based on prior knowledge based on which aspect of either the data, model or algorithm is being enhanced. This taxonomy follows different perspectives on how FSL methods solve the few-shot learning problem. The two, though, generally acknowledge the use of prior knowledge to enhance FSL methods. In this work, we basically follow the approach described in [96], and depicted in Figure 2.3. It is easier to discuss these with recent achievements, challenges, and possibilities of improvement of few-shot learning based deep learning architectures. The literature pertaining to the four approaches to few-shot learning is described in the following sections.

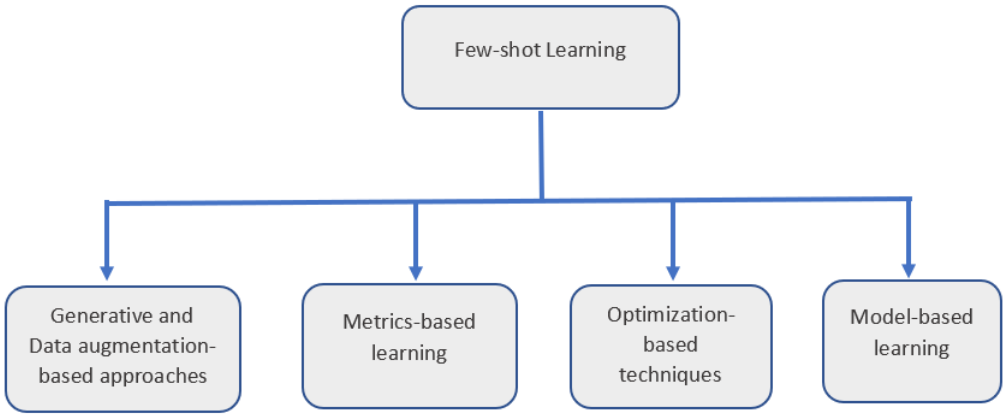


Figure 2.3: Few-shot learning approaches.

2.3.1 Generative and data augmentation-based approaches

In this line of approaches, either generative models [75] are trained to synthesize new data based on few examples of the same class, or by some other form of transfer learning [2, 22, 82, 114, 253, 285]. Generative and data augmentation models create images that have new objects from the same domain that look so close to the real data they are created from. These approaches also take advantage of semi-supervised approaches to generate additional data from unlabelled data [50, 90, 148], either training data synthesis using generative adversarial networks (GANs) [68, 185, 205], and/or augmentation and hallucination of training examples [2, 32, 37, 82, 253] for data-starved classes, among other methods.

In some situations, data can be found in abundance, but labels are highly costly to acquire. For instance, collecting street data is easy and cheap, but high-precision labelling of the photos and video frames is cumbersome, to the point that synthetic datasets [12, 22, 286] are becoming an appealing alternative to real data. Techniques like scaling and rotating, as well as GANs have been implemented to make more extensive the size of the training dataset where the goal is to make the model train and perform better and

generally avoid over-fitting/under-fitting scenarios.

Notable work in data augmentation includes LASO or "Label-Set Operations networks" [2, p. 1] that combines several support set labels of input samples represented by some feature vectors. Samples comprising of set operations of union, or intersection, or/and set-difference are generated from input data samples. The resultant feature vectors will be composed of feature labels that have gone through specific sets of mathematical operations on the example label set of the respective support sets and query sets input.

The work by [82, 253, 285] show that the model's ability to generalize better can be enhanced by conceiving and coming up with almost similar "feature vectors for the training set D_{train}" [82, p. 1] so that the model can be exposed to and trained on additional images. Zhang and Peng [285] uses a saliency network to generate as much as it can the background features and and all the foreground feature information of an image. The idea is to make extensive the prior knowledge by augmenting the available image data samples and generating multiple variety of image samples for model training [12]. Their model uses the generated saliency maps to make better the performance of the few-shot learning technique by learning from the generated images. The model consists of three modules, 1) the saliency network, that generates the saliency maps based on the feature vector of the support samples, 2) a network to encode and mix the background and foreground feature information; and 3) a similarity network.

2.3.2 Metrics-based learning approaches

A number of approaches [70, 103, 113, 171, 172, 174, 196, 214, 216, 224, 239, 247, 255, 260] use a large corpus of instances of known categories to learn an embedding into a metric space where some simple metric is then used to classify instances of new categories via proximity to the few labelled training examples embedded in the same space. In metrics-based learning, the input image samples are modified or reshaped to a lower-level space representation and then classified based on a mathematical distance

metric between the two embeddings, or feature vectors. In other words, an algorithm learns representations from the training samples based on specific objectives. The key question is how to learn the embeddings that are a better representation of the task at hand, or which loss function is good for the intended objective. There are many possible design choices for both the distance functions, including the cosine similarity [162], the Euclidean distance [105, 106] and k-Nearest Neighbours [13]. The basic idea is to learn a distance function between representations of support images typically with a convolutional neural network, and classify query images by comparing them to the labelled support images. Each classification of the query image depends on the distance to the support set images. These metric-based approaches have been used and been able to learn very good embedding spaces with quite meaningful semantics embedded in the metric space[174, 197, 216, 224].

The following section summarises four well known metrics-based approaches: "Siamese networks" [113, p. 1], "Matching networks" [239, p. 1], the "Relation networks" [224, p. 1], and the "Prototypical networks" [214, p. 1]. These attempt to improve the task representative embedding through different architectures and training procedures.

2.3.2.1 Siamese networks

A Siamese network [113], also known as twin neural network (see Figure 2.4), is an architecture with two parallel layers with two convolutional neural networks (CNNs) which are identical and which share similar parameters. The network learns to differentiate between two given inputs by comparing the inputs based on a similarity distance metric, instead of learning to classify using the loss functions. The middle layers extract features of the same kinds from the support set images, as weights and biases are the same. The last layers of Siamese networks use a loss function which calculates the similarity, or difference between the two inputs. Thus, the whole idea of a Siamese network architecture is to learn to discriminate between inputs by using identical CNNs.

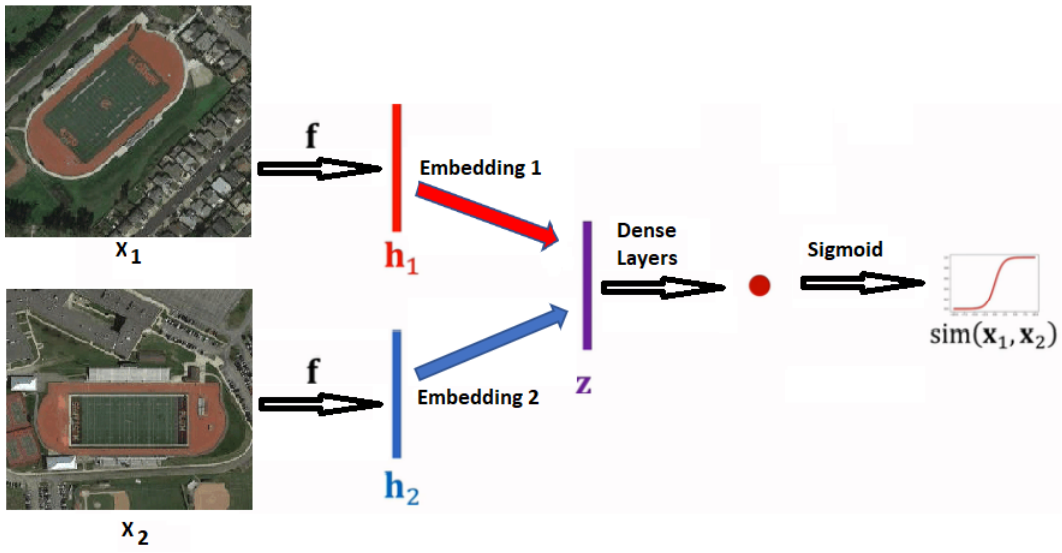


Figure 2.4: An illustration of the basic Siamese network architecture.

For training the Siamese network, pairs of data points X_1 and X_2 have to be created and input. For instance, we can create a pair of similar images, and another pair of dissimilar images. Then we also need to create labels accordingly for similar $(y = 1)$ if the two images have the same feature maps), and dissimilar data points $(y = 0)$, i.e., if the images features are not related at all. Each pair is fed to the Siamese network during training. At the end of the layer, a Siamese network uses a differentiating Contrastive Loss Function [81] which consists of dual terms, the first part is the Mean Squared Error multiplied with their respective labels to decrease the energy of like pairs and, the second part resembles a Hinge Loss [159], with m as a threshold to increase the energy of unlike pairs. The Contrastive Loss Function learns the parameters of the function in such a way that neighbours are pulled together, and non-neighbours are pushed apart [81].

Siamese networks have been widely used in applications that implement one-shot learning such as face detection, fingerprint detection, and signature verification. Some improved versions of the Siamese network architecture have suggested a new loss function known as Triplet Loss [50, 207] which works directly on embedded distances. An

anchor input is compared to a positive input and a negative input. The distance from anchor input to the positive input is minimised, and the distance from the anchor input to the negative input is maximised. The result is that the pair of samples with the same labels are smaller in distance than those with different labels. The loss function penalizes the model such that the distance between the matching examples is reduced and the distance between the non-matching examples is increased.

2.3.2.2 Matching networks

Vinyals et al. [239] (see Figure 2.5) considered that the Siamese neural network was trained on the task of comparison of two images, and differed from the classification task upon which it was evaluated. They then proposed a slightly different version inside of the meta-learning framework that they called Matching networks that implements an end-to-end training procedure that combines feature extraction and differentiable k-nearest neighbour (k-NN) with cosine similarity. Its architecture is majorly inspired by the attention model and memory-based networks. Again, the idea is to map images to an embeddings space, which also encapsulates the label distribution and then project test image in the same embedding space using a different architecture and later, use cosine similarity to measure the similarity metric. They use a comparatively large dataset for solving a task than other few-shot learning approaches. Each image from the support and the query set is fed to a CNN that outputs embeddings for them. The query image is classified using the softmax of the cosine distance from its embeddings to the support-set embeddings. Then the cross entropy loss, or contrast between two variables, on the resulting classification is back-propagated through the CNN. This way, matching networks learn to compute image embeddings, and allow the network to classify images with no specific prior knowledge of classes, achieved simply by comparing different instances of the different classes. Since the classes are different in every episode, matching networks compute features of the images that are relevant to discriminate between classes as op-

posed to standard classification whereby the algorithm learns the features that are specific to each class.

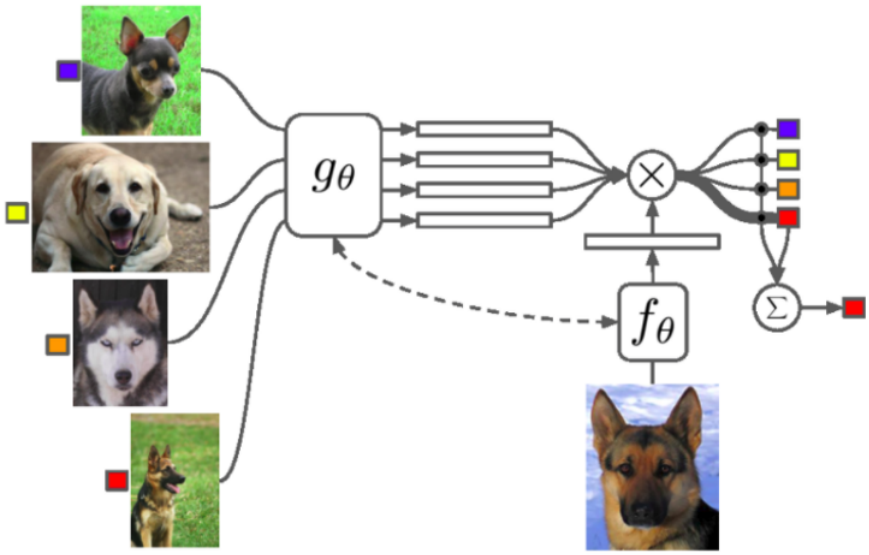


Figure 2.5: Matching networks architecture as illustrated in [239].

The learning of parameters is very slow in matching networks, requiring various weight updates using stochastic gradient descent [202]. When the dataset is small, problems of over-fitting and under-fitting have been encountered, and using regularization and data augmentation have been implemented without solving the problem.

2.3.2.3 Relation network

The "Relation network" [224, p. 1] (see Figure 2.6), trained end-to-end, learns to learn a distance metric to compare a small number of images within episodes in few-shot meta-learning settings. An episode is comprised of indiscriminately chosen support set and query set tasks from the training set D_{train} with k number of annotated labels selected from each class. The network follows a two-steps procedure:

- first step, where the labels from D_{train} and the query set are modified by the embedding module to a lower-level feature representation space;

- second step compares low-level feature representations to measure the similarity between the query image and any of the output class categories by using a relation module.

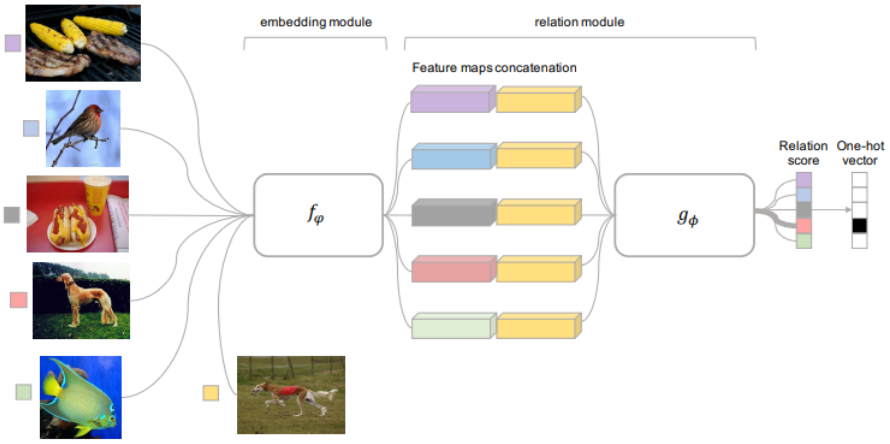
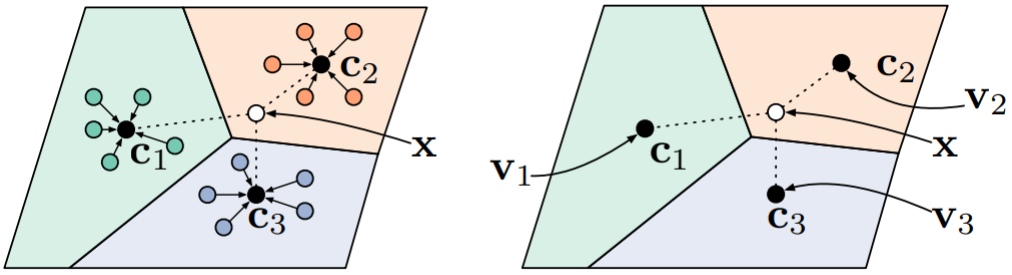


Figure 2.6: Relation networks architecture as illustrated by [224].

2.3.2.4 Prototypical network

The Prototypical Network [214] (see Figure 2.7) is based on the principle that there is always an "embedding in which points cluster around a single blueprint or prototype image feature representation" [214, p. 1] for each class. A neural network is used to learn a non-linear mapping of the input feature space into an embedding space. The class prototype is taken as a mean of its support set feature representations in the embedding space. Each class comes with an embedding of its meta-data into a shared space for giving a high-level feature description of the class rather than a small number of labelled examples. Classification is performed by finding the nearest, depending on the distance, class prototype for an embedded query image point.



(a) Few-shot (b) Zero-shot

Figure 2.7: Prototypical network architecture as illustrated in [214].

A lot of literature is also available about developments in metric learning-based approaches. For instance, the work by [255] proposes an improvement in prototypical networks that learns representations in localisation of realistic settings that results in significant increases in the performance without changing much the model complexity. They deal with class imbalance problems by leave-one-out cross validation. To deal with the clutter problem, they use an learner architecture which can competently confine the image objects features before putting them into specific classes. They employed bilinear pooling to increase the representation power of the learner model, and were able to double the performance results with respect to accuracy of prototypical networks on the meta-iNat [255] benchmark dataset.

Another extension related to the prototypical networks is Learning for Semi-Supervised Classification by [195] that deals with semi-supervised data where data which is not annotated are available together with the supervised training D_{train} samples and their corresponding labels. Together, they can generate prototypes for representations. Consequently, the training D_{train} samples comprises of a tuple **(S, R)**, in which **(S)** is the set of annotated support set image labels and **(R)** is the set of unlabelled image samples. Transferable Prototypical Networks [174] is also based on the rebuilding of the vanilla prototypical network [214] to a network that targets the scenario of unlabelled image samples by jointly extending across the domain between the two. Firstly, the model classifier is

constructed with unlabelled target data with both sources of data and its respective labels. The model then directly predict the target labels of the query or target data.

Oreshkin et al. [172] introduces the task dependent adaptive metric learning that uses different scaling methods. The model generally selects a specific task based on the softmax from the selection of performance measurements. It has a learnable parameter that should allow it to select the best possible metric from the various available collection of metrics. Representative-based metric learning [103] introduces an end-to-end approach that combines multiple models for few-shot object classification and detection. The network trains and learns the network parameters at the same time. It is also designed to learn the feature representation space at that time. Task-Aware Feature Embedding [247] focuses on the construction of feature representations that are set for each particular classification function. They use TAFE-Net which has two modules, the meta learner for learning and producing feature representations for a particular task, and a prediction network for the prediction layer that adjusts to the individual tasks at hand.

Prototypical networks vary from matching networks in the few-shot case with equivalence in the one-shot learning scenario. There is only one support point per class. Therefore, matching networks and prototypical networks become equivalent in such scenarios. The approach is more efficient and far simpler than recent proposed approaches to meta-learning. It has also produced state-of-the-art results even without sophisticated extensions that have been developed for matching networks. This has made prototypical networks an appealing approach to problems of few-shot learning.

2.3.3 Optimization-based techniques

Optimization-based techniques [5, 61, 98, 137, 190, 206, 208, 291] are most understood, and associated with the concept of “meta-learning” [206, p. 1] or “learning-to-learn” [61, p. 1]. They tackle few-shot classification by optimizing model parameters to new tasks, whereby a meta-optimiser is utilised to better train the model so that it can better gen-

eralize during the initial training so that it can provide a better prediction for the novel datasets. In other words, the system focuses on how to converge any objective or loss function instead of minimizing a single loss function, which makes this algorithmic approach task and domain-invariant [96]. For example, to recognize types of flowers using a cross-entropy loss function, one can train the model to learn to understand the difference between any two images, not necessarily flowers, thus making the model task-agnostic. The same model can be used, for instance, for flower recognition and flower detection, and also be domain-agnostic, and used for dog recognition, or any other related task.

Model-agnostic meta-learning (MAML) [61], and its first order MAML [165], attempts to solve the shortcomings of the gradient-descent approach by providing better weight initialization for every new task. The key contribution of MAML is an easy to understand, simple model- and task-agnostic fast learning algorithm. The key idea of this approach is to train the model's parameters using a different dataset. It is then used for novel tasks by using the already initialized parameters to fine-tune the architecture through one or more gradients so that it provides a better performance. The model that they propose can quickly fine-tune the weights by transferring some internal parameters that are more transferable than others. This results in a model that can quickly adapt to novel tasks. This method of training a model's parameters can also be viewed, from a feature-learning standpoint, as building an internal representation. Another variant known as REPTILE [165] is an approximation of MAML that executes SGD for a number of iterations on a given task, and then gradually moves the initialization weights in the direction of the weights obtained after these iterations. The intuition is that every task likely has more than one set of optimal weights, and the goal is to find weight initialisations close to at least one of those optimal weights for every task, and thereafter use the most optimal ones.

Ravi's [190] work is based on LSTM acting as a meta-learner model. The model quickly adapts and update its operational rules and model parameters for training. It

therefore, can generalize and learn a maximisation algorithm which will subsequently be utilised to train a classifier from another model for few-shot learning. The gradient descent [202] algorithm is used for optimizing the network towards a specific task. Jamal [98] considered that a meta model trained on the base dataset (e.g. MAML) could be biased towards some tasks, which potentially results in large variations in the performance on novel tasks. Thus, they proposed a novel TAML ("Task-agnostic Meta-Learning" [98, p. 1]) algorithm which attains comparable performance to the state-of-the-art on 5-way 1-shot and 5-way 5-shot classification on the Omniglot [121] dataset. The algorithm aims to train an initial model that is unbiased to all tasks. During the meta-training process, the task-agnostic property of TAML is established by either maximizing the entropy reduction for each task or minimizing the inequality in performance of various tasks. ES-MAML [217], which is based on evolutionary algorithms attempts to avoid MAML's problem of estimating second derivatives. It is a conceptually simple and easy to implement model, and can handle new types of non-smooth adaptation operators for improving the performance of the model. Other improvements to the original MAML include Multimodal MAML (MMAML) [240]. This improvement has the capability to identify the mode of tasks sampled from a multimodal task distribution and adapt quickly through gradient updates by modulating its prior parameters learnt before according to the identified mode. It allows faster, more efficient adaptation.

Another optimisation-based approach, LEO [204] decouples the gradient-based adaptation procedure from the underlying high-dimensional space of model parameters by learning data-dependent latent generative representation of the model parameters. It therefore performs gradient-based meta-learning in this low-dimensional latent space. WarpGrad (Warped Gradient Descent) [63] methods meta-learn to warp task loss surfaces across the joint task-parameter distribution to facilitate gradient descent. This is achieved by sharing fixed, meta-learned layers across task learners that precondition task parameters during task adaptation.

2.3.4 Model-based approaches

Model-based approaches rely on improved network architectures. These are largely designed with the addition of external memory for the rapid generalization of one-shot learning tasks. They are inspired from how humans store prior information in memory units, and how the information is accessed while learning new objectives. In these approaches, models converge with only a few training steps using information stored in external memory. Some examples include Neural Turing Machines (NTMs) [77], Memory-augmented neural networks (MANNs) [206], and Meta networks [160]. They have not had much impressive results in few-shot learning, and therefore have not received much attention in literature.

NTM (Neural Turing Machines) [77] is inspired by research from the field of computational neuroscience that provide extensive evidence that memory is crucial in the quick and meaningful storage and retrieval of information. An NTM is fundamentally composed of a neural network, consisting of a controller and a two-dimensional matrix known as the memory bank. Each step involves the neural network receiving some input and generating some output corresponding to that input. By so doing, it accesses the internal memory bank and performs read and/or write operations onto it. The basic architecture is shown Figure 2.8.

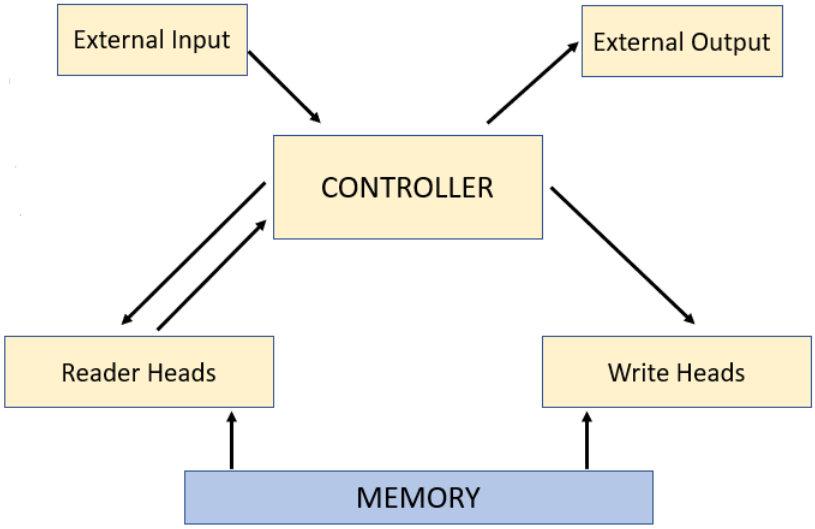


Figure 2.8: Illustration of the architecture of the Neural Turing Machine.

In few-shot settings, the MANN controller uses only content-based addressing, unlike the NTM that uses both content-based addressing and location-based addressing. For a given input, there are only two content-dependent actions a controller might need to take. One action is that the input is very similar to previously seen input in which case we might want to update whatever is in memory. The other action is that current input is not similar to previously seen inputs in which case we do not want to overwrite the recent information but the least used memory location.

In Meta networks [160], a base learner and meta-learner share parameters where a meta-learner extracts prevalent feature embeddings of all tasks in order to acquire a general knowledge of different tasks. The base learner learns the features embeddings of the targeted task. Both these learners framed in a single learner. The knowledge learnt can then be transferred to the base-level learner to provide some generalization in the context of a single, only one task. In Meta networks [160], loss gradients are utilised as meta information to enable models that learn fast weights. The slow and fast weights are then jointly combined to make predictions.

2.4 Few-shot Classification

In deep learning supervised classification, we are interested in learning a model $\hat{y} = \mathrm{f}_\theta x$, parametrised by θ on D_{train} to predict the label $\hat{y} \in 1, \ldots, C_{total}$ for an unlabelled sample x_k on the test set $D_{test} = x_k$, given a dataset $D = \{D_{train}, D_{test}\}$. The training set takes labelled pairs as inputs (x, y), such that $D_{train} = (x_i, y_i)_{i=1}$, $y_i \in 1, ..., C_{total}$, where i is the number of training samples, C_{total} is the number of categories in D_{train}.

Meta-learning methods have received much attention recently, and have been combined with metric-based methods. In few-shot meta-learning, we consider a meta-set $D = \{D_{base}, D_{val}, D_{novel}\}$. C_{base}, C_{val}, C_{novel}, are chosen to be mutually disjoint, where C represents the category. The model learns M on D_{base} that can quickly adapt to unseen categories in D_{novel} with only a few support samples. D_{val} is held-out to assist with adjusting the hyper-parameters during training. A model is evaluated on a set of N-way K-shot classification tasks denoted as $D_{\mathcal{T}} = \mathcal{T}_i$ referred to as *episodes* in few-shot learning settings. Each episode has a split of support set S_i that contains N unique categories with k labelled examples each, and query set Q_i that has the same number of categories, and Q unlabelled examples. For instance, if at test time we are supposed to perform 3-way 2-shot classification, then the training episodes could comprise of $N = 3$, and $K = 2$. An entire episode in few-shot learning is treated as a training instance.

The aim is to learn a classifier to recognise unseen classes during training with limited labelled examples. Meta-learning methods, together with metric learning and augmentations have complemented each other that some authors, e.g. [13, 189, 204] have classified metric-based methods under meta-learning. Similar few-shot learning methods have generally been used in classification, object detection, image segmentation, and other image processing tasks with some modifications. Recent results show that deeper backbones significantly reduce the gap across methods when domain differences are limited.

Some methods, e.g. [28, 46] first learn a deep learning network on all the available images and transfer it to few-shot tasks in test time. Meta-Baseline [30] fine-tunes the

entire network with a nearest-centroid cosine similarity and a scale parameter. Dhillon et al. [46] explores fine-tuning in a transductive setting, where the query set is assumed to be available at the same time. Yu et al. [273] first pre-trained a feature extractor on base-class data, then used the same to initialize the weights of the classifier for the novel classes. The model is updated with a semi-supervised learning method. Lifchitz et al. [138] targeted the transfer of embeddings from a set with abundant data to other sets with few available image examples. This work also proposed attaching new neurons to a previously trained network, or implanting, to learn new, task-specific features that enables training of multiple layers, departing from methods derived from metric learning that train only the final layer. Huang et al. [92] uses fake gradients, and a semi-supervised meta-learning approach that learns from multiple tasks in a transductive environmental setting. They leverage the unlabelled query set in addition to the support set to generate a more powerful model for each task. LGM-Net [127] is also designed to learn transferable prior knowledge across various tasks. It then directly produces network parameters for similar unseen tasks with training samples. It has two fundamental modules, the first called TargetNet being a neural network for solving a specific task. The second one is called MetaNet and it aims at learning to generate functional supportive weights for the TargetNet module by observing the initial training samples.

Recent techniques [30] focus on generalising to unseen domains at test time in Meta-Dataset [234]. CNAPS [10] uses the Mahalanobis distance, class-covariance-based distance metric and adopts a non-parametric classifier. SNAIL [157] combines temporal convolutions to aggregate information from past experience, and soft attention to pinpoint specific pieces of information. The work by Gidaris et al. [72] extends an object detection and recognition model with an attention-based weight generator for few-shot classification. They redesigned the classifier of a ConvNet model as the cosine similarity function between feature representations and classification weight vectors to learn novel representational categories from only a few training data while at the same time remem-

bering the base categories on which it was trained. This process unifies the detection and recognition of both novel and base representations. It also improve recognition of feature representations that generalize better on unseen object categories. Qi et al. [182] used weight imprinting to recognize novel visual categories to ConvNet classifiers by directly setting the final layer weights from novel training examples during low-shot learning. The weight imprinting process directly sets weights for a new category based on an appropriately scaled copy of the embedding layer activations for that training example. Qiao et al. [183] adapts a pre-trained neural network to novel categories by directly predicting the parameters from the activations. Zero training is required in adaptation to novel categories, and fast inference is realized by a single forward pass. VAGER [291] generalizes meta-learns in the concept space rather than in the complicated instance space. TAFE-Net or "Task-Aware Feature Embedding Networks" [247, p. 1] adapts the image representation to a new task in a meta learning fashion. The network model consists of a meta-learner that generates parameters for the feature layers in the prediction network so that the feature embeddings can be accurately adjusted for that task.

MetaSGD [137] proposes a SGD like optimiser [5] that has a much higher representation perfomance by learning to learn the learner initializations and the learner update direction and learning rate simultaneously. All the learning takes place in a single meta-learning process. The model adapts easily and quickly to the various novel tasks. It is easily applicable for both supervised learning and reinforcement learning. Learning involves two steps. The first step involves the meta-learner gradually learning on the different tasks in the meta-space. The second step is based on the feedback of the meta-learner where the learning approach of the meta-learner is evolved in the learning space.

Few-shot learning methods that have used weights from pre-trained classification models, e.g. ResNet or EfficientNet family pre-trained on a large-scale datasets have generally been better than randomly initialised ones. For that reason, [223] used weights of the deep neural network for transfer learning using the operations of scaling and shift-

ing to indicate how to transfer. Some classification methods [183, 203, 204] have also used pre-trained weights, with the weights fine-tuned for each classification task. A recent model by [1] propose a metric-learning loss for minimizing the distance between related base samples and the centroid of novel instances in their feature representations. It also has a conditional adversarial alignment loss based on the Wasserstein distance. This leverages part of the base data by aligning the novel training instances to the closely related ones in the base training set.

Most of these methods do not consider the time and resource efficiency which limits their practical use. They also depend on hyper-parameter tuning on each specific dataset. There are real-world scenarios with generally unknown datasets and tasks. More so, many datasets always change over time. The two other main challenges that make it difficult from making a fair comparison among few-shot classification algorithms are the discrepancy of the implementation details among multiple few-shot learning algorithms, and the performance of baseline approaches. The performance of baseline approaches can be significantly under-estimated when, for instance, some models are trained without data augmentation. While the current evaluation focuses on recognizing novel classes with limited training examples, these novel classes are sampled from the same dataset. The lack of domain shift between the base and novel classes makes the evaluation comparison scenarios unrealistic.

2.5 Object Detection

Object detection is a fundamental problem in computer vision in which the objective is to obtain the objects' specific positions in the input image, and classifying each object according to each type. In image classification, there is usually only one main target object in the image and the model's sole focus is to identify the target category. However, in many situations, there are multiple targets in the image that we are interested in. The

task of object detection therefore, involves two subtasks, 1) localizing one or more objects within an image, and 2) classifying each object in the image. Therefore, the model predicts the class of the image like in image classification tasks, and also predicts the coordinates of the bounding box or mask that fits the detected object. The conventional object detection framework in deep learning usually has four components:

- Region proposal, i.e., a deep learning model is used to generate regions of interest (ROI) to be further processed by the system.
- Feature extraction and network predictions, the pre-trained CNN network that is used for feature extraction to extract features from the input image that are representative for the task at hand and use these features to determine the class of the image.
- Non-maximum suppression (NMS), to avoid repeated detection of the same instance by combining overlapping boxes into a single bounding box for each object.
- Evaluation metrics, e.g. mean average precision (mAP), precision-recall curve (PR curve), and intersection over union (IoU).

Two approaches have mainly been utilised in deep learning-based object detection systems, 1) a dual-step object detector, and 2) a one-step object detector. Region-based convolutional neural networks (R-CNN) series [73, 74, 198] represent a two-step object detector to propose area of interest called a Region-of-Interest (RoI) and clarify the RoI with classification and localisation. The original R-CNN uses a first network to determine the ROI in an image, and another following network to classify the content of each ROI. The later variations, the Fast R-CNN and later the Faster R-CNN have tried to make the algorithm work better by reducing the number of ROI, as well as to lessen the redundant computations on the image.

Single-stage object detectors including YOLO family [192, 193, 194] and SSD [146] and their variations detects objects in a single forward pass directly using a single fully

CNN. The prediction of the bounding-box and the label of each object are done concurrently. These one-step detectors require far less computational complexity since they are proposal free methods. These have gone through incremental improvements since their creation. SSD improves YOLO by employing default boxes (anchors) to adjust to various object shapes. YOLOv2 improves YOLO with a series of techniques such as multi-scale training and new network architecture (DarkNet-19). Proposal-free methods do not require a per-region classifier, which makes them significantly faster. Recently, YOLOR ("You Only Learn One Representation" [241, p. 1]) that encodes implicit knowledge and explicit knowledge together was proposed. It can learn knowledge from normal learning as well as subconsciousness learning, and can learn a unified representation to integrate the two, and simultaneously serve multiple tasks, including object detection, panoptic segmentation, multi-label image classifications among the many tasks.

2.5.1 Few-shot object detection

The model for "few-shot object detection" [55, p. 1] achieves learning-to-learn by generally transferring feature embeddings or representations that can be generalised to recognise novel objects given a few data samples of training images and their corresponding labels. Prior works to few-shot object detection have generally been formulated in three paradigms. Initial works used transfer learning [24] via fine-tuning given a feature extractor trained on a dataset with abundant base classes. Other notable methods that address the problem using transfer-learning are the fine-tuning approach (TFA) [248], and Multi-scale Positive Sample Refinement (MPSR) [256]. Distance-metric learners, for instance, RepMet [103] extracts meta-level knowledge representations that easily and quickly adjust to new class instances by learning on other auxiliary tasks. The output class instances are firmly controlled on support set images which are commonly employed for few-shot learning.

Others re-weight full image features using class-specific attention vectors as in You

Only Look Once (YOLO) [101], or utilise RoI features, for example Faster-RCNN [263], while [54, 55] use attentive feature vectors together with mathematical relational operators [224] to learn lower-level feature embeddings that are able to recognise the base input categories from the novel class categories. In [259], the authors demonstrate the use of additional feature representations to further guide the object detection network model, an approach further expanded by [250] to separate the learning between class-specific and class-agnostic components. These models can also resolve misclassification or mask identification issues between categories in the predicted RoIs.

Few-shot metric-learning approaches [55, 134, 153, 164, 187, 208, 259] learn an applicable feature representation space in which features of same class examples are similar. The features of mismatched classes are therefore categorised as different, or unrelated. They have been able to learn embedding in some cases with quite meaningful semantics embedded in the metric [101]. In Δ-encoder [208], learning to synthesize samples of categories unseen during training when only a single or a few real examples are available. The encoder learns to extract transferable deformations between pairs of examples of the same class, while the decoder learns how to apply these deformations to other examples to learn to sample from new categories. Fan et al. [55] propose a matching metric between image pairs based on the Faster R-CNN framework equipped with attention RPN and multi-relation detector trained using the contrastive training strategy. They contributed a new highly diverse FSOD ("Few-shot Object Detection" [55, p. 1]) dataset that contains 1000 categories of various objects with high-quality annotations. The same idea was extended to [164] and [153] implemented using GANs [75] to learn a deep image embedding on unlabelled data with two loss functions, a reconstruction loss, and the triplet loss aimed at self-supervised learning. Rahman et al. [187] proposed a unified any-shot detection model that utilises a rebalanced loss function. It uses semantics as prototypes for object detection, a formulation that naturally minimizes knowledge forgetting and mitigates the class-imbalance in the label space. The model can concurrently learn to detect

both zero-shot and few-shot object classes. Another model by [259] is composed of a feature-extractor, a feature attention highlight module as well as a two-stage detection back-end that can quickly adapt to novel classes. The performance of metric learners is in many cases comparable to the meta-learning approaches that have also been used for object detection in few-shot settings.

Meta-learning approaches [44, 66, 102, 109, 109, 129, 145, 152] have also been used in object detection. These are designed to learn a meta-learner to parameterise the optimization algorithm, resulting in models that once trained can learn on new such tasks with relatively few examples and adapt to new environments quickly. Deng et al. [44] and Kang et al. [101] redefine one-stage YOLOv2 [192] by applying re-weighting of the features scheme to an object detector. They also readdress a two-stage Faster R-CNN [198] object detector with the assistance of a meta-learner that inputs support set images together with bounding box annotations. In [44], a re-weighting module effectively learns to extract meta-feature representation knowledge from the support set images, and adaptively assign different weights for each feature representation from the support images. It also uses a bounding box prediction module that executes the object detection task on the re-weighted feature maps of the support images based on YOLOv3 [193]. The model by [102] also has a meta feature learner that extracts features from labelled base classes, and a re-weighting module within a one-stage detection architecture using a YOLOv2 framework (i.e., DarkNet-19). The re-weighting module M, taking the support examples as input, learns to embed support information into re-weighting vectors and adjust contribution of each meta feature of the query image accordingly for following metric approaches. In [109] a prototypical feature knowledge transfer supported with an attached meta-learning model is proposed. The meta-learner's input are support set images that are composed of the few examples of the novel categories and those of the base categories. The model predicts prototypes that represent each category as a vector embedding. Then, the prototypes re-weight each ROI feature vector from a query image to remodel R-CNN

predictor heads. They predict the prototypes under a graph structure. The Meta-SSD [66] is composed of a meta-learner and an object detector. The meta-learner can teach the detector how to learn from few examples in just one updating step by utilising a Single-Shot MultiBox Detector (SSD) [146] employed as the object detector. Meta-RetinaNet [129] is trained by the Balanced Loss and employs a Meta Coefficient Learner (MCL) to augment the deep neural networks. The MCL adapts to tasks for all the convolutional layers by employing the product of pre-trained convolution weights and coefficient vectors. It could adequately transfer the learned knowledge to new tasks while overcoming the over-fitting problem by training fewer parameters.

Among the seminal work for FSOD include LSTD or "Low-Shot Transfer Detector for Object Detection" [24, p. 1]. LSTD design a flexible deep architecture to alleviate transfer difficulties in low-shot detection that integrates the advantages of both SSD and Faster R-CNN in a unified deep framework. Second, they introduce a regularized transfer learning framework where the transfer knowledge and background depression regularizations are proposed to leverage object knowledge respectively from source and target domains, in order to further enhance fine-tuning with a few target images. Meta R-CNN [263], Towards General Solver for Instance-level Low-shot Learning also extends Faster Mask R-CNN by proposing meta-learning over RoI (Region-of-Interest) features instead of a full image feature which disentangles multi-object information merged with the background, enabling Faster R-CNN/Mask R-CNN turn into a meta-learner to achieve object detection tasks. They specifically introduce a Predictor-head Remodelling Network (PRN) that shares its main backbone with Faster R-CNN/Mask R-CNN. The input to PRN are images containing low-shot objects together with their bounding boxes or masks to infer their class attentive feature vectors. These feature vectors take channel-wise soft-attention on RoI features, re-modelling those R-CNN predictor heads to detect and/or segment the objects consistent with the classes these vectors represent.

MetaYOLO [101] or "Few-shot Object Detection via Feature Re-weighting" [101,

p. 1], and MetaDet [252] or "Meta-Learning to Detect Rare Objects" [252, p. 1] both leverage meta-level knowledge from fully-labelled base classes. They easily and quickly adapt to novel classes by using a meta feature learner for extracting meta features that are generalizable to detect novel object classes by using training data from base classes with sufficient samples. They also use a re-weighting module within a one-stage detection architecture. The re-weighting module transforms the support examples from the novel classes to a global vector that indicates the relevance of meta features for detecting the corresponding objects. These two modules use a carefully designed loss function. They are, together with a detection prediction module, jointly trained end-to-end based on an episodic few-shot learning. FSDetView [259] propose a meta-learning framework that can be applied to the tasks of few-shot object detection and few-shot viewpoint estimation including for 3D data by leveraging on rich feature information originating from base classes with many samples. They propose a simple joint feature embedding module to make the most of this feature sharing.

MPSR ("Multi-Scale Positive Sample Refinement for Few-Shot Object Detection" [256, p. 1]) tackles the problem of scale variations to enrich object scales in FSOD. It generates multi-scale positive samples as object pyramids and refines the prediction at various scales, and can be integrated as an auxiliary branch to Faster R-CNN with Feature Pyramid Network(FPN) that have been used with object detection systems for building high-level feature maps of an input image at several image scales, delivering a strong FSOD solution.

FSOD ("Frustratingly Simple Few-Shot Object Detection" [55, p. 1]) improved the object detection tasks by "fine-tuning only the last layer of existing detectors" [246, p. 1] to achieve incredible results to the "few-shot object detection" [55, p. 1] task, and outperformed meta-learning methods. However, the high variance in the image samples frequently results in unreliability of existing benchmarks. They therefore, had to examine and make corrections their evaluation methods by involving multiple groups of training

labels, and eventually obtained substantial improvements in terms of comparisons, and managed to build new benchmarks based on PASCAL VOC, COCO [140] and LVIS [80] datasets.

FSCE ("Few-Shot Object Detection via Contrastive Proposal Encoding" [221, p. 1]) present an approach to learning good feature embeddings by contrastive-aware object encodings via a contrastive proposal encoding loss (CPE loss). This facilitates the classification of identified objects from the image by promoting instance level intra-class compactness and interclass variance. SRR-FSD or "Semantic Relation Reasoning for Shot-Stable Few-Shot Object Detection" [293, p. 1] employs "semantic relation together with the visual information" [293, p. 1]. Each class concept is represented by a representation learned from a large image dataset. The detector is trained to undertake image representations of objects into this meaningful embedding space. Fan et al. [55] proposed Attention-RPN, Multi-Relation Detector and Contrastive Training strategy. These exploit the similarity between few shot support set and query set to detect novel objects. It also suppresses false detection in the background. A new dataset with high-quality annotations called FSOD with 1000 categories of diverse objects is also introduced with this work.

"Meta R-CNN" [263, p. 1] and its variations [136, 256, 258, 259] are built upon "Faster R-CNN" [198, p. 1]. These meta-learn channel-wise attention layer for remodelling the RoI head. ONCE [178] and Meta-YOLO [101] are grounded on single-stage detectors. These Meta-learning detectors usually need initially well-located regions. To get an initial well-located region is usually hard to to obtain without learnable shape priors and fine-tuned RPN, especially if the dataset has very few training images. FSOD [55] makes an effort to do away with this issue by "meta-learning an Attention-RPN" [55, p. 1]. MetaDet [251] leverages meta-level knowledge about model parameter generation for category-specific components of novel classes. Other approaches, e.g. [107] to Object detection with limited labelled samples has been addressed in weakly-supervised settings . They consider bounding box annotations to be expensive to obtain, and therefore consider

the problem of training object detectors with only image-level labels.

The recently proposed GenDet [145] is trained by numerous few-shot detection tasks sampled from base classes each with sufficient samples, and thus it is expected to generalize well on novel classes. An adaptive pooling module is further introduced to suppress distracting samples and aggregate the detectors generated from multiple shots. The algorithm trains a reference detector for each base class in the conventional way, with which to guide the training of the detector generator. The reference detectors and the detector generator can be trained simultaneously. Finally, the generated detectors of different classes are encouraged to be orthogonal to each other for better generalization.

Kim et al. [109] further developed the idea by introducing prototypical network [214] knowledge transfer into "few-shot object detection" [55, p. 1] which is premised on the belief that there exist embeddings in which similar class points cluster around a single prototype representation. They attached a meta-learner that takes support set images that include the few labels of the novel image classes and base classes from the dataset. They predict unique prototypes under a graph structure that represent each class category as a vector. Then, the prototypes are used to re-weight each ROI feature vector from a query image. This is done to remodel R-CNN predictor heads.

2.6 Knowledge Distillation

Knowledge distillation [18, 89, 181] is a model compression [18] method in which a smaller model is trained to imitate a larger model, or ensemble of models that have been pre-trained, and the training has to be done without loss of validity (see Figure 2.9 for a generic model framework). Work by [89, 181] demonstrate convincingly that the feature knowledge acquired by a large ensemble of models can be transferred to a smaller model through learning. The performance is generally accomplished by minimizing a "loss function in which the target is the distribution of class probabilities predicted by

the teacher model, or the output of a softmax function on the teacher model's logits" [89, p. 4]. Since the softmax does not provide much information beyond the ground truth labels already provided in the dataset, [89] introduced the concept of "softmax temperature" [p. 1] to tackle this issue. The probability π of class is calculated from the logits z. Extensive downstream computer vision tasks, such as semantic segmentation, transfer learning, image classification, and object detection, can significantly benefit from the distilled pre-trained models.

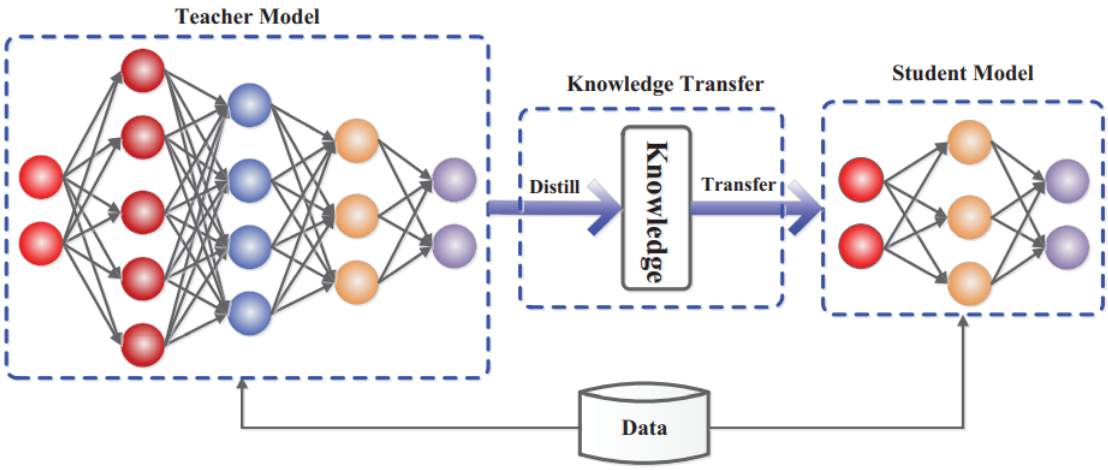


Figure 2.9: Generic teacher-student framework for knowledge distillation. Image source [76]

In their experiments, Hinton [89] use temperature values ranging from 1 to 20. Lower temperatures work better when the student model is very small compared to the teacher model. As the temperature is raised, the resulting soft-labels distribution becomes richer in information. There is normally no way to predict the capacity for information the smaller model will contain after training. Experiments indicate that a very small model might not be able to capture all of the information from the bigger model. In their experiments, Hinton et al. [89] use a weighted average between the distillation loss α and the student loss β. They obtained the best results when setting α to be much smaller than β. Other works [23, 120, 295] which utilize knowledge distillation do not use a weighted

average. Some set $\alpha = 1$ while leaving β tunable, while others do not set any constraints.

Building supervised learning models has been widely used as machine learning techniques that work effectively in performing regression and classification tasks. They require data to be manually labelled. This process is generally slow, error prone, and expensive. It also suffers from such issues as "generalization error, spurious correlations, and adversarial attacks" [144, p. 1] which slows down model building. To avoid these limitations, self-supervised learning has recently gained momentum in an effort to eliminate the need for data labels. It aims at embedding augmented versions of similar samples closer and diverse samples far from each other, achieved by a similarity metric to measure the closeness of the two embeddings [97]. By building models autonomously, supervised learning can be employed without any external interaction, and can effectively mimic how humans come up with certain decisions by using their own intellect. It can largely reduce the cost and time to build machine learning models. It aims to address challenges in supervised learning when it comes to collecting comprehensive data, cleaning, classify, and labelling data for clear embeddings, a process which is arduous and time-consuming compared with how humans approach learning.

Self-supervised learning [47, 222, 275] can be viewed as an autonomous form of supervised learning though it is some form of unsupervised learning since there is no manual label involved. It does not require human input in the form of data labelling. Whereas unsupervised learning concentrates on detecting specific data patterns, such as clustering, or anomaly detection; self-supervised learning aims at recovering, which is still in the paradigm of supervised settings [145]. Self-supervised learning method involves two steps: pre-training the network with unlabelled data, and training on the target task with labelled data as a downstream task.

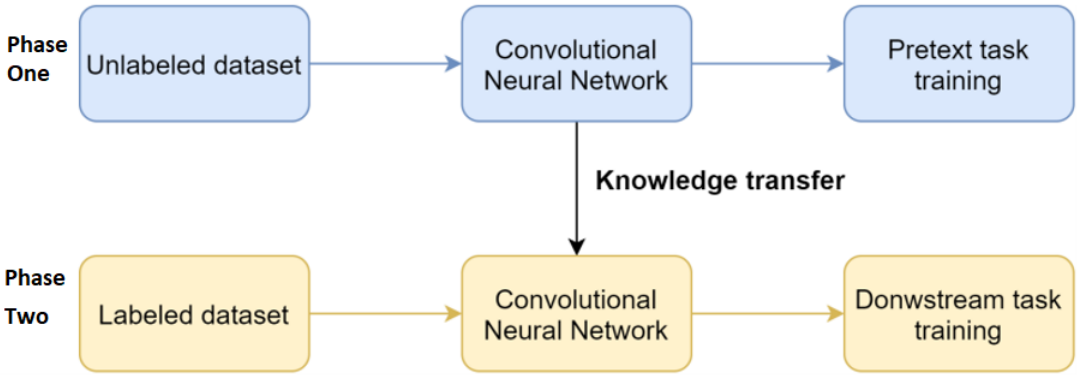


Figure 2.10: The general pipeline of self-supervised learning. An unlabelled dataset is used to pre-train the network that is then used to train the network on labelled dataset.

A common workflow for self-supervised representation learning on images is to train a model with unlabelled images and then use one intermediate feature layer of this model to feed a multinomial logistic regression classifier. Some recent work propose training supervised learning on labelled data and self-supervised pretext tasks on unlabelled data simultaneously with shared weights [222, 275]. Rotations, or other argumentations of an entire image [71] is another interesting way for self-supervised learning. This modifies an input image while the semantic content of the image remains unchanged.

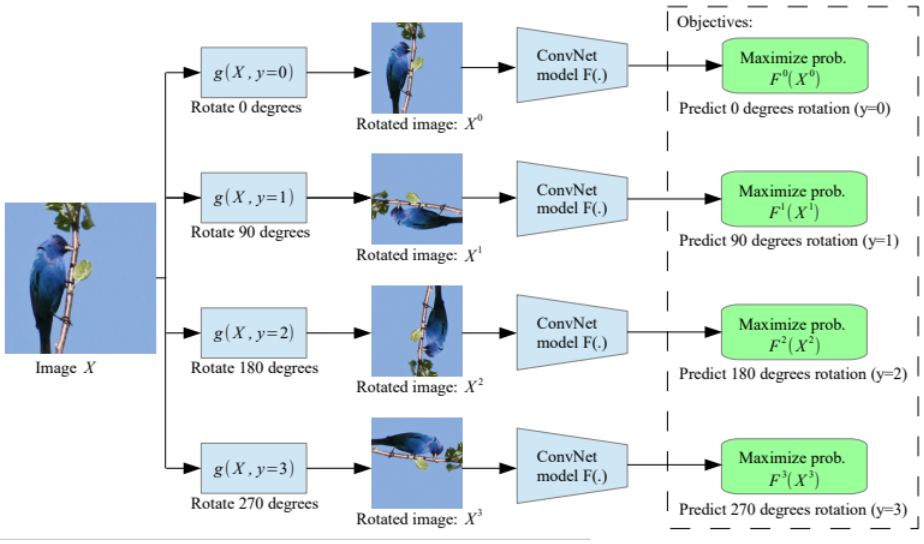


Figure 2.11: Illustration of self-supervised learning by rotating the input images. (Image source: [71])

Doersch et al. [47] formulates the self-supervised task as predicting the relative position between two random patches from one image. This way, the model needs to understand the spatial context of objects on the image to tell the relative position between constituent parts. Noroozi and Favaro [168] followed the idea of chromatic aberration to design a jigsaw puzzle game as pretext task: the model is trained to place nine shuffled patches back to the original locations that are each independently processed by a convolutional network with shared weights and outputs a probability vector per patch index out of a predefined set of permutations. Colourisation [281] has also been proposed, where a model is trained to colour a grayscale input image. The task is to map this image to a distribution over quantized colour value outputs in the CIE Lab* colour space to approximate human vision.

The GANs have also been able to learn to map from simple latent variables to arbitrarily complex data distributions. For example, [49] introduced bi-directional GANs with an additional encoder to learn the mappings from the input to the latent variable.

The discriminator predicts in the joint space of the input data and latent representation to tell apart the generated pair from the real one. Van Den Oord et al. [236] proposed the Contrastive Predictive Coding (CPC) that translates a GAN modelling problem to a classification problem. The idea of momentum contrast that stores representations of all the data points in a database and samples a random set of keys as negative examples was used in [85, 257]. Chen et al. [27] proposed a "framework for contrastive learning of visual representations that learns representations for visual inputs by maximizing agreement between differently augmented views" [27, p. 1] of the same sample via a contrastive loss in the latent space.

Bootstrap Your Own Latent (BYOL) [78] relies on two neural networks, referred to as online and target networks with the same architecture that learn from each other. They managed to accomplish state-of-the-art performances in the absence of negative samples. CURL (Contrastive Unsupervised Representations for Reinforcement Learning) [218] relies on random crop data augmentation in reinforcement learning that trains a visual representation encoder by ensuring that the embeddings of the augmented versions match by using a contrastive loss. It learns a visual representation for RL tasks by "matching embeddings of two data-augmented versions and of the raw observation via contrastive loss" [218, p. 1].

Dvornik et al. [52] design an "ensemble of deep networks to leverage the variance of the classifiers" [52, p. 2], and introduce strategies to encourage the networks to cooperate, while encouraging prediction diversity. Lee et al. [125] trains the model to learn a single unified task with respect to the joint distribution of both the original and self-supervised tasks fully-labelled datasets without optimising the summation of their corresponding losses. The original image labels are augmented via self-supervision of input transformations. They also propose a knowledge transfer technique they call self-distillation. Li et al. [135] introduced a two-stage procedure to learn a multi-domain networks by distilling knowledge of multiple separately trained networks after co-aligning their features

with the help of adapters and centred kernel alignment which can be further refined for previously unseen domains by utilising distance learning methods.

Zhao and Wen [288] present a novel two-phase pipeline that leverages self-supervised learning and knowledge distillation by first learning a teacher model which possesses rich and generalizable visual representations via self-supervised learning, and secondly to distil the representations into a student model in a self-distillation manner by fine-tuning the student model for image classification. A margin loss for the self-supervised contrastive learning proxy task is also proposed. Rajasegaran et al. [188] also follows a 2-stage process for learning whereby a neural network is trained to make as large as possible the entropy of the feature embedding. They utilise a self-supervised loss in the metric space. In the Stage Two, they create an output manifold, and then minimize the entropy on feature embedding by bringing self-supervised twins together. Their work inhibits the manifold with student-teacher knowledge distillation. Self-Supervised Knowledge Distillation (SSKD) [269] sought to improve the "quality of labels by capturing feature representation from multiple augmented views" [269, p. 3] of unlabelled image using two modules: 1) the identity learning, that explores the relationship between "unlabelled samples and predicts their one-hot labels by clustering to give exact information for confidently distinguished images and the soft label learning, and soft label learning regards labels as a distribution and induces an image to be associated with several related classes for training peer network in a self-supervised manner " [269, p. 1]. Rizve [199] propose a training mechanism that jointly "enforces equivariance and invariance to a general set of geometric transformations" [199, p. 1].

Roy et al. [201] leverage the intra-class knowledge from the neighbour classes with the intuition that neighbour classes share similar statistical information. A regressor is trained on the "many-shot classes and is used to evaluate the few-shot class means from a few samples, then superclasses are clustered to obtain each superclass' statistical mean and feature variance to be used as transferable knowledge inherited by the children few-

shot classes" [201, p. 6]. Such knowledge is then used by a generator to "augment the sparse training data" [201, p. 7] to help the downstream classification tasks. Wang et al. [246] employed the visual knowledge to help the feature extractors focus on different visual parts, and design a classifier to learn the distribution over all input categories. They then develop three schemes to minimize the prediction error and balance the training procedure, i.e. hard labels (a label assigned to a member of a class where membership is binary) to provide precise supervision, semantic textual knowledge is utilized as weak supervision to find the potential relations between the novel and the base categories, and an imbalance control is presented from the data distribution to alleviate the recognition bias towards the base categories.

Momentum [2] Teacher [135] present a student-teacher based self-supervised learning that performs momentum update on both network weights and batch normalization (BN) statistics. The teacher's weight is a momentum update of the student, and the teacher's BN statistics is a momentum update of those in history. Chen et al. [30] use of a relatively deep and wide networks during unsupervised pre-training and supervised fine-tuning on a few labelled examples, and found out that the fewer the labels, the more this approach (task-agnostic use of unlabelled data) benefits from a bigger network. After fine-tuning, the big network can be "further improved and distilled into a much smaller one with little loss in classification accuracy by using the unlabelled examples for a second time" [30, p. 3], but in a task-specific way.

Cui et al. [40] performs distillation by only driving prediction of the student model consistent with that of the teacher model instead of frameworks which require student model to be consistent with both soft-label generated by teacher model and hard-label annotated by humans. Koohpayegani et al. [115] follows the same approach by developing a model compression method to compress an already learned, deep self-supervised teacher model to a smaller student one to mimic the relative similarity between the data points in the teacher's embedding space.

In contrast, [227] introduce a self-supervised learning algorithm that use a soft similarity for the negative images rather than a binary distinction between positive and negative pairs. They iteratively distil a slowly evolving teacher model to the student model by capturing the similarity of a query image to some random images and transferring that knowledge to the student using contrastive learning (see Figure 2.12). Following the same path, Adversarial Contrastive Learning (ACL) [100] leverages contrastive learning framework learning representations by maximizing feature consistency under differently augmented views, and integrating self-supervised pre-training with adversarial training to improve robustness-aware self-supervised pre-training by learning representations that are consistent under both data augmentations and adversarial perturbations.

Results of ongoing research [30, 135, 201, 227] in self-supervised knowledge distillation using few-shot learning methods indicate that these methods can significantly reduce the training time and cost of neural networks. It can also assist in downstream computer vision tasks such as object detection and image classification that can significantly benefit from the distilled pre-trained models.

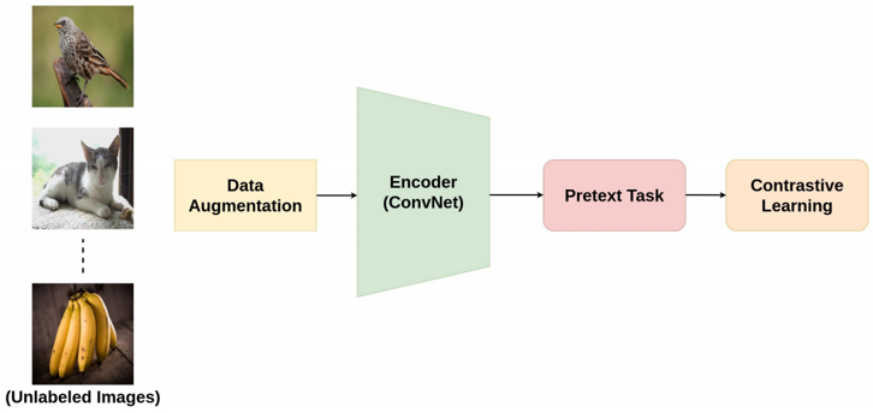


Figure 2.12: Contrastive learning for self-supervised learning. Image source: [71]

2.7 Image Segmentation in Deep Learning

Image segmentation can be formulated as the problem of classifying pixels with semantic labels, also known as semantic segmentation; or partitioning of individual objects, or instance segmentation, or both semantic and instance segmentation, referred to as panoptic segmentation. Semantic segmentation performs "pixel-level labelling with a set of object categories for all image pixels" [284, p. 3]. Instance segmentation extends the scope of semantic segmentation by detecting and delineating each object of interest in the image.

Early work on image segmentation methods include thresholding, k-means clustering, histogram-based methods, region-growing, watershed methods, active contours, Markov random fields, and sparsity-based methods. In recent years, deep learning models have caused a paradigm shift in the field. A new generation of image segmentation models with remarkable performance improvements, often achieving the highest accuracy rates on popular benchmarks (e.g. Figure 2.13) have been developed. Image segmentation has become a key computer vision and image processing with important applications such as augmented reality, medical image analysis, scene understanding, and video surveillance, among others. Various architectures have been developed, including encoder-decoder architectures, visual attention models, multi-scale and pyramid-based approaches, convolutional pixel-labelling networks, recurrent networks, and generative models among others.

Figure 2.13: Some examples of qualitative segmentation results of DeepLabV3 [244] on sample images.

Several deep learning models have been proposed in literature for instance and semantic segmentation, including fully convolutional networks [150], who used transfer learning to modify VGG16 and GoogLeNet to output partial segmentation maps instead of classification scores. They used skip connections (Figure 2.14) in which "feature maps from the final layers of the model are up-sampled and fused with feature maps of earlier layers" [150, p. 1]. The model combines semantic information (from deep, coarse layers) and appearance information (from shallow, fine layers) in order to produce accurate and detailed segmentations. Tested on PASCAL VOC, NYUDv2, and SIFT Flow datasets, the model achieved state-of-the-art segmentation performance. Various works have demonstrated that fully-connected networks can be applied to such areas as brain tumour segmentation [282], instance-aware semantic segmentation [66, 229], skin lesion segmentation [274] and iris segmentation [138] in an end-to-end fashion. These models have computationally been expensive for real-time inference, and are not generalisable to 3D images.

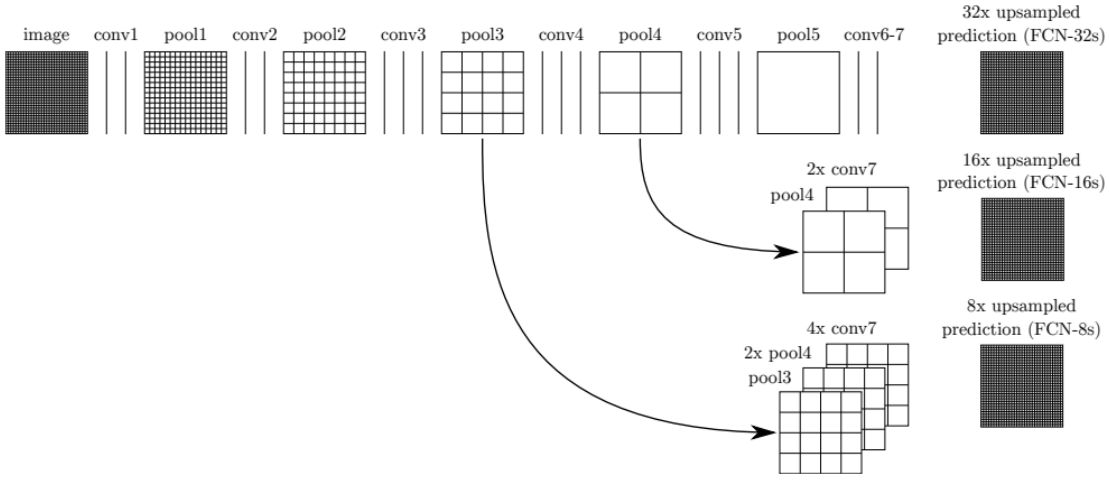


Figure 2.14: Illustration of skip connections for segmentation. Image source [150]

Noh et al. [167] introduced a model for semantic segmentation based on transposed convolution, or deconvolution. Their model (see Figure 2.14), consists of an encoder using convolutional layers adopted from the VGG 16-layer network, and a multilayer deconvolutional network that inputs the feature vector and generates a map of pixel-accurate class probabilities. The latter comprises deconvolution and unpooling layers, which identify pixel-wise class labels and predict segmentation masks. SegNet ([7]) (Figure 2.15), a fully convolutional encoder-decoder architecture with a decoder upsamples its lower-resolution input feature maps using pooling indices computed in the max-pooling step of the corresponding encoder to perform non-linear up-sampling. A limitation of the encoder-decoder based models in image segmentation is the loss of fine-grained image information, due to the loss of resolution through the encoding process.

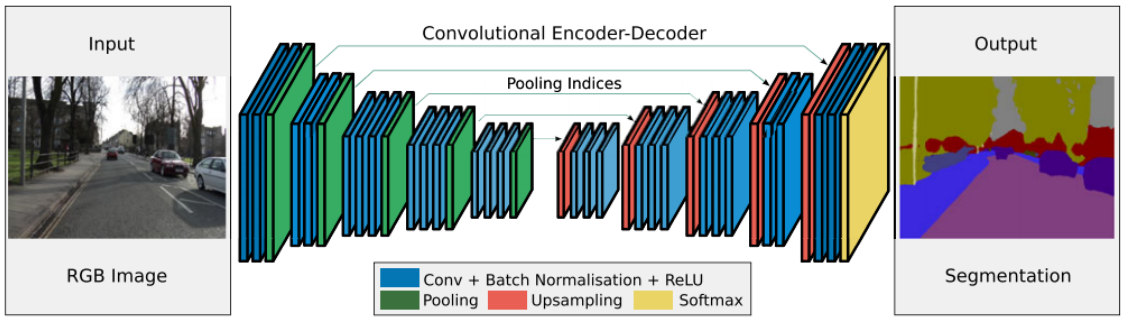


Figure 2.15: An illustration of the SegNet architecture which is fully convolutional. Image source [7]

U-Net [200] (Figure 2.16), proposed for segmenting biological microscopy images, was trained on 30 transmitted light microscopy images. The architecture comprises two parts, a contracting path to capture "context, and a symmetric expanding path that enables precise localization" [200, p. 1]. Its training strategy relies on the use of data augmentation to learn effectively from very few annotated images.

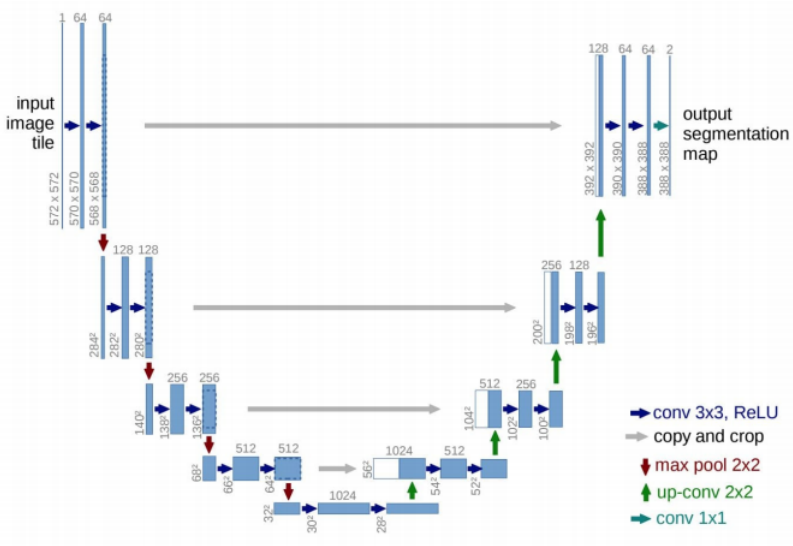


Figure 2.16: The U-Net semantic segmentation model on sample images. Image source [200]

Models that use R-CNN (Region-Based Convolutional Neural Networks) such as [73, 74, 198] have proven successful in object detection applications by using a region proposal network (RPN) that proposes bounding box or mask candidates that extracts a Region of Interest (RoI), and an RoIPool layer computes features from these proposals to infer the bounding box coordinates and class of the object have been used to address the instance segmentation problem by simultaneously performing object detection and semantic segmentation [83].

Other models that have been proposed are based on RNNs [179], Attention-Based Models [79], Generative adversarial networks [75]. All the above models have only been successful when trained with large datasets.

2.7.1 Few-shot semantic and instance segmentation

Instance segmentation, e.g. using Mask R-CNN [84] aims to "discriminate objects in the pixel level, which is a finer representation compared with detected boxes, and can be subdivided into box-based methods" [184, p. 2], e.g. [20, 111, 184, 261] that utilize detected boxes to locate objects, and box-free approaches [34, 248, 264] that generate instances without the assistance of object boxes. Semantic segmentation assigns each pixel with a semantic category, "without considering diverse object identities " [261, p. 1]. The task of few-shot segmentation [209] aims at assigning a category label to each image pixel with few annotated samples. The dense prediction can only be achieved under the guidance of latent features defined by sparse annotations. Few-shot segmentation methods perform image segmentation for a particular object class in a query image, using a small set of support image-mask pairs. Just like image classification, successful training of a semantic segmentation model have required large densely-annotated supervised learning image datasets that are costly to obtain [129, 144, 270]. Once the the model is done with training, new object categories are difficult to add to the model, meaning that the model has to undergo new training.

In few-shot settings, which is the focus of this work, the aim is to train a model that can quickly adapt to new tasks given a few examples for input. In this few-shot learning task, there is a densely annotated training dataset D_{train} which consists of objects in base categories C_{train}. The model aims to train based on the training set and evaluate it on a testing set D_{test}, both consisting of novel object categories C_{test}, i.e. $C_{train} \bigcap C_{test} = 0$. The testing set D_{test} is specifically constructed in an episodic form — for a K-shot learning task, each episode $e_i = \{(\mathbf{S}_i, Q_i)\} x_s^k$ and y_s^k consists of a support set $S_i = \{(x_s^k, y_s^k)\}, k \in \{1 \dots K\}_i$ and a query set $\mathbf{Q}_i$ = $\{(x_q, y_q)\}$ x_s^k and y_s^k $\mathbf{x}_s^k$ and y_s^k $_i$, where x_s^k and y_s^k are the k^{th} support image and its corresponding object mask, respectively. x_q and y_q are the query image and the ground truth, respectively. During each testing episode, the model is asked to perform segmentation on x_q based on the object information in x_s^k and y_s^k.

Shaban et al. [209] was the original work that tackled the problem of few-shot image segmentation. Their method directly "predicts the weight of the dense-classifier based on support images" [209, p. 1]. They also created a dataset, namely Pascal-5^i [209] for few-shot segmentation which has become one of the most used benchmark for evaluating few-shot segmentation methods. Subsequent works on few-shot semantic segmentation are typically based on a two-branch comparison framework, which can be seen as an extension of metric learning methods in few-shot image classification. Wang et al. [246] tackled the few-shot segmentation problem from a non-parametric metric learning perspective and present PANet that learns class-specific prototype representations from support images for each semantic class within an embedding space and then performs segmentation over the query images through matching each pixel to the learned prototypes. CANet [276] is a class-agnostic segmentation network that performs few-shot segmentation on new classes that consists of a "two-branch dense comparison module which performs multi-level feature comparison between the support image and the query image, and an iterative optimization module which iteratively refines the predicted re-

sults" [276, p. 1]. Furthermore, they introduce an attention mechanism to effectively fuse information from multiple support examples under the setting of k-shot learning. The encoder network is used to map the low resolution encoder feature maps to full input resolution feature maps for pixel-wise classification, and resembles the 13 convolutional layers in the VGG16 network [226]. The decoder uses "pooling indices computed in the max-pooling step of the corresponding encoder to perform non-linear upsampling" [276, p. 1] thereby eliminating the need for learning to upsample.

Other ideas [143, 148] use prototypical representation by learning to extract prototypes from both support and query images of the known classes. Liu et al. [148] decompose the class representations into a set of "part-aware prototypes" [148, p. 1], capable of capturing "diverse and fine-grained object features" [148, p. 1]. In addition, they leverage unlabelled data to enrich the same part-aware prototypes, resulting in better modelling of intra-class variations of semantic objects. They also develop a novel graph neural network model to generate and enhance the proposed part-aware prototypes based on labelled and unlabelled images. Liu et al. [144] propose a Prototype Refinement Network (PRNet) that learns to bidirectionally extract prototypes from both support and query images of the known classes. They use an adaptation step that makes the model learn new concepts which is directly implemented by retraining, and prototype fusion which fuses support prototypes with query prototypes, incorporating the knowledge from both sides.

Inspired by few-shot classification work by [61] and [190], [176] proposed a Class-Agnostic Few-shot Edge detection Network (CAFENet) based on meta-learning strategy they called "few-shot semantic edge detection" [176, p. 1], aiming to localize crisp boundaries of novel categories. It employs a "semantic segmentation module" [p. 1] in small-scale to compensate for lack of semantic information in edge labels. The predicted segmentation mask is used to "generate an attention map to highlight the target object region, and make the decoder module concentrate on that region" [176, p. 1]. They also propose a new regularization method based on multi-split matching, and two new

datasets, FSE-1000 and SBD-5 I for semantic edge detection. Hendryx et al. [87] proposed EfficientLab architecture to evaluate first-order model agnostic meta-learning algorithms, including REPTILE [165] on few-shot image segmentation, and used Bayesian optimization to infer the optimal test-time adaptation routine hyperparameters. He et al. [83] proposed a framework called Mask R-CNN for object instance segmentation that extends Faster R-CNN [198] by adding a branch for predicting an "object mask in parallel with a branch for bounding box recognition" [198, p. 1]. The approach detects objects in an image while simultaneously generating a segmentation mask for each instance. Fan et al. [55] extends the Mask R-CNN [83] to produce a Fully Guided Network (FGN) that perceives few-shot instance segmentation as a guided model where the support set is encoded and utilized to guide the predictions of a base instance segmentation network. In this view, FGN introduces different guidance mechanisms into the various key components in Mask R-CNN, including "Attention-Guided RPN, Relation Guided Detector, and Attention-Guided FCN" [55, p. 1], in order to make full use of the guidance effect from the support set and adapt better to the inter-class generalization.

Various other approaches have been explored. Tian et al. [228] formulated the few-shot segmentation problem as a learning-based pixel classification problem, and propose a framework they called MetaSegNet can be trained by the episodic training mechanism, and is based on meta-learning whose architecture of embedding module consisting of the global and local feature branches is developed to extract the appropriate meta-knowledge. Siam et al. [211] propose a method that constructs the new class weights from few labelled samples in the support set, while updating the previously learned classes [211]. They extended the work on adaptive correlation filters inspired from the work on an adaptive masked imprinted weights. Their method utilizes a "masked average pooling layer on the output embeddings" [211, p. 3] that acts as a positive proxy for that class. It is then used to adaptively update the 1x1 convolutional filters that are responsible for the final classification. Tian et al. [232] proposed the "Prior Guided Feature Enrichment

Network (PFENet), designed to have a training-free prior mask generation method and Feature Enrichment Module (FEM) that overcomes spatial inconsistency" [232, p. 2] by adaptively enriching query features with support features and prior masks.

Similarity Guidance Network [286] is proposed as an end-to-end framework for one-shot segmentation by predicting the segmentation mask of a query image with the reference to one densely labelled support image of the same category. To obtain the robust representative feature of the support image, they first adopt a "masked average pooling strategy for producing the guidance features by only taking the pixels belonging to the support image" [286, p. 2] into account. They then leverage the cosine similarity to "build the relationship between the guidance features and features of pixels" [286, p. 2] from the query image. In this way, the possibilities embedded in the produced similarity maps can be adapted to guide the process of segmenting objects. Zhao et al. [290] tackle one-shot semantic segmentation problem by first training an object-ness segmentation module which generalizes well to unseen categories. Then the object-ness module is used to predict the objects present in the query image, and train an "object-ness-aware few-shot segmentation model that takes advantage of both the object information and limited annotations of the unseen category" [290, p. 2] to perform segmentation in the query image. Zhu et al. [293] presents an adaptive tuning framework, in which "the distribution of latent features across different episodes is dynamically adjusted based on a self-segmentation scheme, augmenting category-specific descriptors" [293, p. 2] for label prediction. Specifically, a novel "self-supervised inner-loop is firstly devised as the base learner to extract the underlying semantic features from the support image. Then, gradient maps are calculated by "back-propagating self-supervised loss through the obtained features, and leveraged as guidance for augmenting the corresponding elements" [293, p. 2] in embedding space. Finally, with the ability to continuously learn from different episodes, an "optimization-based meta-learner is adopted as outer loop of their proposed framework to gradually refine the segmentation results" [293, p. 2].

Stacked Deconvolutional Network (SDN) [66] propose a model in which multiple shallow deconvolutional networks are stacked one by one to integrate contextual information and bring the fine recovery of localization information, with "inter-unit and intra-unit connections designed" [66, p. 2] to assist "network training and enhance feature fusion" [66, p. 2] since the connections improve the flow of information and gradient propagation throughout the network. Hierarchical supervision is applied during the upsampling process of each SDN unit to enhance the discrimination of feature representations and optimization. Gairola et al. [67] demonstrate gaps in the utilization of similarity information in few-shot segmentation in existing methods, and propose SimPropNet, that jointly predict the support and query masks to force the support features to share characteristics with the query features. They also utilize similarities in the background regions of the query and support images using a novel foreground-background attentive fusion mechanism. Li et al. [129] proposed a few-shot segmentation dataset, FSS-1000, which consists of 1000 object classes with pixel-wise annotation of ground-truth segmentation.

2.7.2 Panoptic segmentation

Kirillov et al. [112] propose the challenging task of panoptic segmentation that unifies the typically distinct tasks of semantic segmentation and instance segmentation [112] into one which is our main focus in this part of the thesis. They also propose a novel "panoptic quality (PQ) metric" [112, p. 1] that captures performance for all classes "('stuff' and 'things')" [112, p. 1] in an interpretable and unified manner. The goal in panoptic segmentation is to perform a unified segmentation task. A 'thing' is a countable object, for example animal, table or tree. It is a category that has instance-level annotation, the same used in object detection. The 'stuff' is amorphous, structureless region of similar texture such as road and sky. It is a category without instance-level annotation. Studying 'thing' comes under object detection and instance segmentation, while studying 'stuff' comes under semantic segmentation. Encoding pixels involves assigning each pixel of an image

two labels, one for semantic label, and other for instance identity. The pixels having the same label are considered belonging to the same class, and instance identity for 'stuff' is ignored. Unlike instance segmentation, each pixel in panoptic segmentation has only one label corresponding to one instance, i.e. there are no overlapping instances.

Panoptic segmentation segments the image $I \in \mathbb{R}^{H \times W \times 3}$ into a cluster of categorised masks for the whole image:

$\{y_i\}_{i=1}^{K} = \{(m_i, c_i)\}_{i=1}^{K}$,

where K represents the ground truth masks $m_i \in \{0,1\}^{H \times W}$ do not coincide or encroach into each other, and c_i denotes the prior terrestrial observations that have been made into ground truth class labels of mask m_i.

Many approaches for panoptic segmentation extend Mask R-CNN [84] for object instance segmentation that extends Faster R-CNN [198] by adding a branch for predicting an object mask in parallel with a branch for bounding box recognition. For instance, Fully Guided Network (FGN) [55] perceives few-shot instance segmentation as a "guided model where the support set is encoded and utilized to guide the predictions of a base Mask R-CNN instance segmentation network" [55, p. 1]. It introduces different guidance mechanisms into the various key components in Mask R-CNN, including what they called "Attention-Guided RPN, Relation-Guided Detector, and Attention-Guided FCN, in order to make full use of the guidance effect from the support set" [55, p. 1] and adapt better to the inter-class generalization. Efficient Panoptic Segmentation (EfficientPS) [158] incorporates a novel semantic head that "aggregates fine and contextual features coherently and a new variant of Mask R-CNN as the instance head" [158, p. 1]. They also propose a novel "panoptic fusion module" [158, p. 1] that congruously integrates the output logits from both the heads of their architecture to yield the final panoptic segmentation output. Additionally, they introduce the "KITTI panoptic segmentation dataset" [158, p. 1] that contains panoptic annotations for the KITTI benchmark.

Other notable work include Panoptic-FCN [129], that aims to represent and predict

foreground 'things' and background 'stuff' [112] in a unified fully convolutional pipeline. Specifically, Panoptic-FCN encodes each object instance or 'stuff' category into a "specific kernel weight with the proposed kernel generator and produces the prediction by convolving the high-resolution feature directly" [129, p. 1]. With this approach, "instance-aware and semantically consistent properties for 'things' and 'stuff' can be respectively satisfied in a simple generate-kernel-then segment workflow" [129, p. 1]. De Geus et al. [42] propose a single deep neural network for panoptic segmentation, that makes joint semantic and instance segmentation predictions and combines these to form an output in the panoptic. The entire prediction is made in one pass, reducing the required computation time and resources. Ying et al. [270] proposed an end-to-end encoder-decoder network architecture that utilises object information from support samples to separate target objects from the background in a query image. They design an object representation generator (ORG) module incorporated into the architecture to aggregate local object features from support images and produce object-level representation. The ORG module can be embedded into the network and trained end-to-end in a weakly-supervised fashion without extra human annotation.

Occlusion Aware Network (OANet) [142] predict both the instance and 'stuff' segmentation in a single network. Moreover, they introduce a novel "spatial ranking module to deal with the occlusion problem" [127, p. 1] between the predicted instances. DeeperLab [266] present a "single-shot, bottom-up approach for panoptic segmentation that generalizes the tasks of semantic segmentation for 'stuff' classes and instance segmentation for 'thing' classes" [266, p. 1], assigning both semantic and instance labels to every pixel in an image. Axial-DeepLab [244] predicts pixel-wise offsets to pre-defined instance centres. These centre-based proxy sub-task makes it difficult to deal with objects with irregular shapes. MaX-DeepLab [243] directly projects labelled masks with a "mask transformer", and learns using a "panoptic quality inspired loss" [243, p. 1] via "bipartite matching" [243, p. 1]. The mask transformer employs a "dual-path architecture that

introduces a global memory path in addition to a CNN path" [243, p. 1]. This allows direct exchange of information with all CNN layers. The approach simplifies the pipeline that depends heavily on "surrogate sub-tasks and hand-designed components, such as box detection, non-maximum suppression, and thing-stuff merging" [243, p. 1]. Panoptic-DeepLab [34] adopts the "dual-ASPP (Atrous Spatial Pyramid Pooling) and dual-decoder structures" [34, p. 1] specific to semantic and instance segmentation, respectively. The semantic segmentation branch is the same as the DeepLab [266], while the instance segmentation branch is class-agnostic, involving a simple instance centre regression.

CIAE (Category- and Instance-Aware Pixel Embedding) [69] simplifies the panoptic segmentation pipeline by consistently modelling the two classes with a novel panoptic segmentation framework with a pixel-wise embedding feature that encodes both semantic-classification and instance-distinction information, and which extends a detection model with an extra module to predict category- and instance-aware pixel embedding. CondInst (conditional convolutions for instance and panoptic segmentation) [231] unifies instance and panoptic segmentation. It designs dynamic instance-aware mask heads, conditioned on the instances to be predicted instead of using instance-wise ROIs as inputs to the instance mask head of fixed weights.

Bounding-Box Free Network (BBFNet) [17] predicts coarse watershed levels and uses them to detect large instance candidates [17] where boundaries are well defined. For smaller instances, whose boundaries are less reliable, BBFNet also predicts "instance centres by means of Hough voting followed by mean-shift" [17, p. 1] to reliably detect small objects. A novel triplet loss network helps merging fragmented instances while refining boundary pixels. The approach is differs from prior works in panoptic segmentation that rely on a combination of a semantic segmentation network based on Mask R-CNN to guide the prediction of instance labels, which is costly computationally for instance segmentation. They use this observation to exploit class boundaries from semantic segmentation networks and refine them to predict instance labels.

BANET [30] introduce a novel deep panoptic segmentation scheme based on a bidirectional learning pipeline, and a plug-and-play occlusion handling algorithm to deal with the occlusion between different object instances. Chen et al. [30] introduce a novel deep panoptic segmentation scheme based on a bidirectional learning pipeline, and a plug-and-play occlusion handling algorithm to deal with the occlusion between different object instances. UPSNet [261] proposes a panoptic head that does not use parameters. This allows backpropagation to both panoptic segmentation modules, while DETR [20] generally depends on the box prediction and detection.

Weber et al. [254] uses an object detector to design an end-to-end single-shot method that segments both countable object instances as well as background regions into a non-overlapping panoptic segmentation at almost video frame rate. The model has a shared encoder-decoder backbone, and utilises multiple branches for semantic segmentation, object detection, and instance centre prediction. The panoptic head combines all outputs into a panoptic segmentation and can even handle conflicting predictions between branches as well as resolving inter- and intra-class overlaps to achieve a non-overlapping segmentation. SPiNet [94] integrated execution flows for semantic and instance segmentation and generated a unified feature map they called Panoptic-Feature with information on 'things' and 'stuff'.

2.7.2.1 Panoptic quality

Panoptic quality(PQ), or the product a recognition quality (RQ) term and a segmentation quality (SQ) term, measures the "quality of a predicted panoptic segmentation relative to the ground truth" [112, p. 1], and involves segment matching and "PQ computation given the matches of 'stuff' and 'things' in a uniform way" [112, p. 1], thus unifies the evaluation over all classes. It is normally calculated per class independently, and the results averaged over classes, making it sensitive to class imbalances.

The Intersection over Union (IoU) (see Figure 2.17) is a common way to describe the

quality of the model's bounding box prediction over an object on the image. By diving the area the prediction and ground truth intersect by the area they both consume, we get a ratio that is inclusively between 0 and 1, with 0 meaning there is no intersection and 1 being a perfect fit.

$$IoU = \frac{Prediction \cap GroundTruth}{Prediction \cup GroundTruth}$$

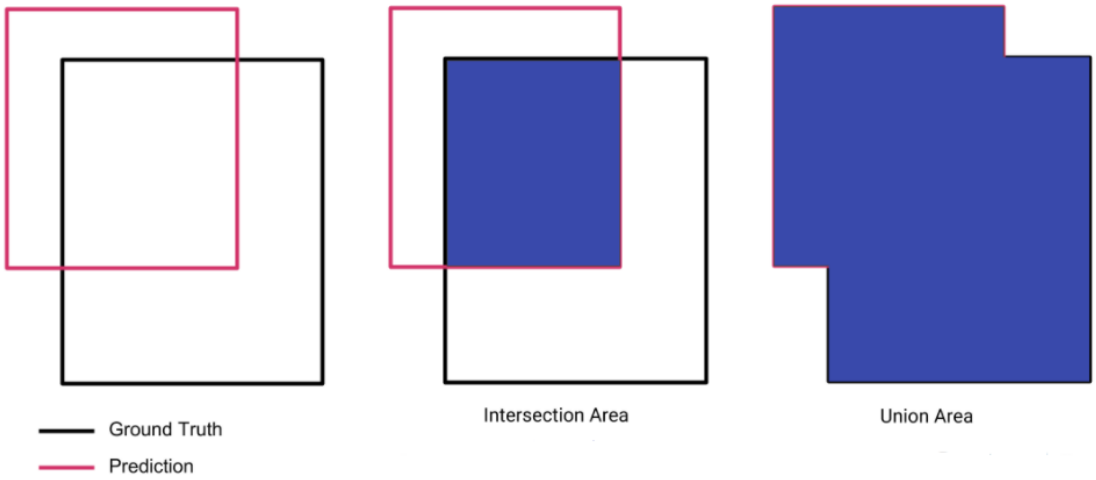


Figure 2.17: Illustration of Intersection over Union (IoU).

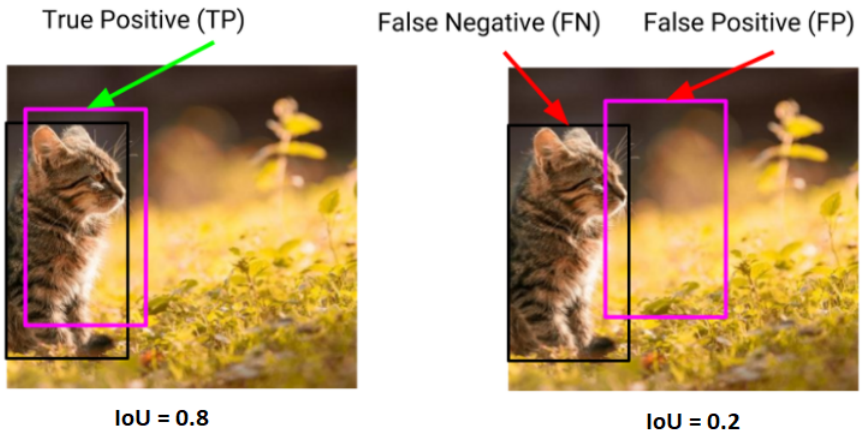


Figure 2.18: Illustration of TP, FN and FP

For each class in a dataset, the unique matching splits the predicted and ground truth segments into three sets: true positives (TP), false positives (FP), and false negatives (FN), representing matched pairs of segments, "unmatched predicted segments, and unmatched ground truth segments" [112, p. 1], respectively (see Figure 2.18). PQ is defined as:

$$PQ = \frac{\sum_{(p,q)\in TP^{IoU}(p,q)}}{|TP|+\frac{1}{2}|FP|+\frac{1}{2}|FN|}$$

$\sum_{(p,q)} \in TP^{IoU}(p,q)$ is the average IoU of matched segments, while

$\frac{1}{2} \mid FP \mid + \frac{1}{2} \mid FN \mid$ is added to the denominator to penalize segments without matches

2.8 The Vision Transformer

The Transformer [237] has recently emerged as an alternative to CNNs for visual recognition [51, 233, 241, 289]. The Vision Transformer (ViT) [51] (see Figure 2.19) is an attention-based neural network which has become the de facto standard for natural language processing. In computer vision, attention has been applied in conjunction with CNN layers, or to replace certain components of CNNs, or replace CNNs entirely. An image is split into linearly embedded fixed-size non-overlapping patches, positional embeddings added, and the resulting sequence fed into transformer encoder (see Figure 2.20). Recently, they have emerged as an alternative to CNNs [20, 51], and a pure transformer has been applied directly to sequences of image patches, e.g. in [51], the first paper that successfully trains a transformer encoder on ImageNet, attaining very good results compared to familiar convolutional architectures. The image patches are expected to be of the same size. Although they are competitive to CNNs in terms of their global computations and perfect memory, they have yet to be used extensively [20]. They are currently more computationally expensive, and require much more data to train than CNNs [21].

The architecture of the ViT follows the following basic steps:

- Split an image into patches, flatten the patches,
- Produce lower-dimensional linear embeddings from the flattened patches, add positional embeddings,
- Feed the sequence as an input to a standard transformer encoder,
- Pre-train the model with image labels (fully supervised on a huge dataset), and
- Fine-tune on the downstream dataset for image classification.

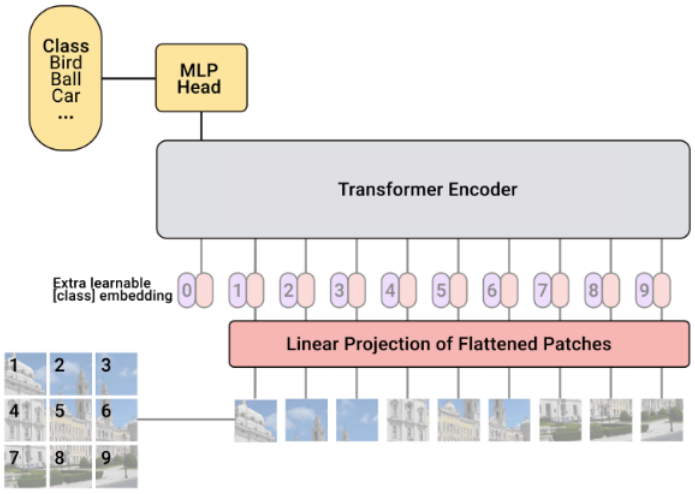


Figure 2.19: Illustration of the Vision Transformer. The image is split into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. Image source [51].

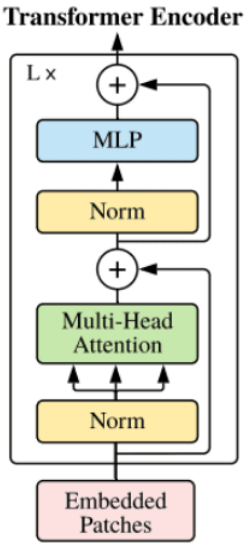


Figure 2.20: The Transformer encoder block is identical to the original transformer proposed by [237].

Although [51] is the first paper that directly applies a pure transformer to sequences of images and show that the reliance on CNNs is not necessary, there have been previous attempts in image processing. Notable work includes those that combine CNNs with some forms of self-attention by augmenting feature maps, e.g. by [11], or by further processing the output of a CNN using self-attention, for example for object detection and panoptic segmentation [20]. Chen et al. [30] trains a model in an unsupervised fashion as a generative model, and applies transformers to image pixels after reducing the image resolution and the colour space, thereby achieving a 72% accuracy on the ImageNet dataset.

DETR (DEtection TRansformer) [20] consists of a convolutional backbone followed by an encoder-decoder transformer which can be trained end-to-end for object detection and panoptic segmentation. It removes the need for hand-designed components unlike models such as Faster-R-CNN and Mask-R-CNN, which use region proposals, non-maximum suppression procedure and anchor generation as preprocessing steps. Deformable DETR [293] mitigates the high complexity and slow convergence issues of

DETR via a novel sampling-based efficient attention mechanism. Its attention modules only attend to a small set of key sampling points around a reference, and can achieve better performance than DETR especially on small objects, and with less training epochs.

The CrossTransformer [48] is a few-shot transformer-based neural network architecture that takes a small number of labelled images and an unlabelled query, and then infers class membership by computing distances between spatially-corresponding features. It uses prototypical networks blueprint that aggregate information in a spatially-aware way, using local features that are more likely to generalize. They demonstrated a good performance on the Meta-dataset [234]. The classifier is more robust to task and domain shift.

The Swin Transformer [149] is a hierarchical transformer with the flexibility to model at various scales, whose representation is computed with shifted windows. This brings greater efficiency by limiting self-attention computation to non-overlapping local windows while allowing cross-window connections. It has linear computational complexity with respect to image size, making it compatible with a broad range of computer vision tasks. DINO [21] showed the potential of self-supervised pre-training a standard ViT model to developing a BERT-like [45] model that have been successful with language processing.

2.9 Datasets for Few-Shot Learning

Few-shot learning usually adopts episodic training. In order to sample different tasks for the few-shot learning classification, a dataset such as Omniglot [122, 239], tieredImageNet [195], MiniImageNet [190, 239], CIFAR-FS [15, 116], FC100 [116, 172], and CUB 200 [88], with many different classes is required. The models are expected to learn from each of the episodes. Most of the current few-shot learning algorithms are benchmarked on some of these few-shot learning datasets. Omniglot is a dataset for handwritten

characters from 50 different alphabets that consists of 1623 samples in total [121, 122]. Each character is a 105 x 105 greyscale image. There are only 20 samples for each character, each drawn by a distinct individual. MiniImageNet is a lighter version of the original ImageNet dataset, designed specifically for evaluation of the few-shot learning models. This dataset consists of 100 classes with 600 samples of 84 × 84 colour images for each class. CUB 200 is a dataset of photos of 200 different bird species with 6,033 samples in total. Many other datasets have recently been proposed for use for benchmarking few-shot learning algorithms, including Meta-Dataset [234] that has been used for benchmarking with meta-learning. The TCGA [277] is a dataset of classification tasks over the values of an attribute based on the gene expression data from patients diagnosed with specific types of cancer. FSOD models have been evaluated on the datasets that have been used for few-shot learning. Pascal 5^i [209] and COCO-20^i [140, 163] have widely been used benchmarks for "few-shot object detection" [55, p. 1] and object classification.

A variety of datasets have been used for panoptic segmentation training and testing, including Cityscapes [39], Mapillary Vistas [161], MS COCO [140], Wild Panoramic Panoptic Segmentation (WildPPS) [99], KITTI panoptic segmentation dataset [158, 195]. Pascal-5^i [209] and COCO- 20^i [163] are widely-used benchmarks. The COCO dataset contains "80 'thing' classes and 53 'stuff' classes" [163, p. 2], with $118K$, $5K$, and $20K$ images for training, validation, and testing, respectively. Cityscapes dataset consists of "$5,000$ street-view fine annotations with size 1024 × 2048" [39, p. 1], which can be divided into $2,975,500$, and $1,525$ images for training, validation, and testing, respectively. Mapillary Vistas is a "traffic-related dataset with resolutions ranging from 1024 × 768 to more than 4000×6000" [161, p. 1]. It includes 37 'thing' classes and 28 'stuff' classes with $18K$, $2K$, and $5K$ images for training, validation, and testing (p. 1), respectively. Other two recent datasets that have been popular for few-shot semantic and instance segmentation are the Cityscapes-Panoptic-Parts and PASCAL-Panoptic-Parts introduced by [154]. Both have annotations compatible with panoptic segmentation, and additionally,

they have part-level labels for selected semantic classes. In this work, we will use a number of few-shot learning datasets, including the Pascal-5^i [209], which was created based on Pascal VOC 2012, with 20 categories in the original PASCAL-VOC dataset evenly divided into 4 splits for 4-fold (1 split for testing and the other 3 splits for train) cross-validation, and COCO-20^i [163] from MS-COCO 2014 (see Table 4.2) for benchmarking. We will also benchmark with the Omniglot, Mapillary Vistas, and Oxford Flowers 102. The more challenging MS COCO-20^i has 80 categories in the original MS-COCO 2014 dataset that are "evenly divided into 4 splits for 4-fold cross-validation (20 categories for testing and the remaining 60 categories for training, and 1000 support-query pairs for testing in each split)" [163, p. 2].

2.10 Conclusion

This chapter provides a comprehensive review of some of the most relevant literature in few-shot learning, including the literature related to few-shot learning in computer vision tasks of image classification, object detection, knowledge distillation and segmentation. Work on meta-learning, metric learning, data augmentation, panoptic segmentation, and self-supervised learning was also explored, with more emphasis on work related to the few-shot learning environment.

In Chapter 3, we propose an alternative novel method for few-shot classification method by employing novel dual meta-learners with a meta-ensemble module for generalisation and inference of images.

Chapter 3

Few-Shot Image Classification with Dual Meta-Learners

3.1 Introduction

In the previous chapter, a comprehensive study of the literature, including the models and algorithms that have been proposed for few-shot image classification, object detection, knowledge distillation and image segmentation in deep learning supervised settings was explored. Several approaches, including generative and augmentation-based approaches, metric learning-based learning, meta-learning and optimisation-based learning were explained. In this chapter, we introduce a novel meta-learning model for few-shot classification that consists of dual meta-learners supervised by a central controller that controls a feature extraction module and a meta-learning module, and a meta-ensemble module for integrated inference and generalisation. Each meta-learner is composed of a pre-trained encoder fine-tuned by batch training and parameter-free decoder used for prediction. We use ResNet-152 as a backbone to learn vector representations f_θ of input and ImageNet pre-trained weights. We then optimize the classifier by using the cosine distance with a learnable scale parameter in the feature space in the meta-training stage. Empirical

evaluation on the Omniglot, Oxford Flowers102 and MiniImageNet datasets is provided.

3.2 Meta Learning

In supervised learning settings, meta-learning learns from a set of labelled tasks, each represented as a labelled support and query set, and a testing set [61, 137]. A meta-learner is a "trainable learning algorithm that can train a learner by influencing its actions or behaviour" [61, p. 1]. By being exposed to a broad scope of a task space, a meta-learner may figure out a learning strategy tailored to the tasks in that space, and learn gradually. A meta-learning algorithm or "learning-to- learn" [61, p. 1] is expected to improve its performance with a number of increased training episodes by carrying out rapid learning within each task, whose feedback is used to adjust the learning strategy of the meta-learner [61, 238, 239]. The base-learner, one of the components in meta-learning, works at the "level of individual tasks, or episodes" [206, p. 1] which in few-set settings is characterised by having a small set of labelled images. The meta-learner learns on a bunch of similar tasks to maximize the combined generalization from such sequential episodes with the goal of improving the performance of the base learner. The learning process can "continue forever" [206, p. 1], thus enabling continual or "life-long learning" [3, p. 1], and at any moment, the meta-learner can be applied to learn a learner for any new task [59]. Meta-learning has been applied successfully to few-shot learning on classification [28, 138], object detection [134, 136, 153] and reinforcement learning [180, 218].

Metric based, e.g., prototypical networks [103, 131, 172] and optimization based, e.g., MAML [5, 61, 137] methods have been the most popular and effective methods that have been used with meta-learners (see Chapter 2 for the detailed literature). They both do not learn an explicit learner [61], which is typically done by an optimizer such as SGD.

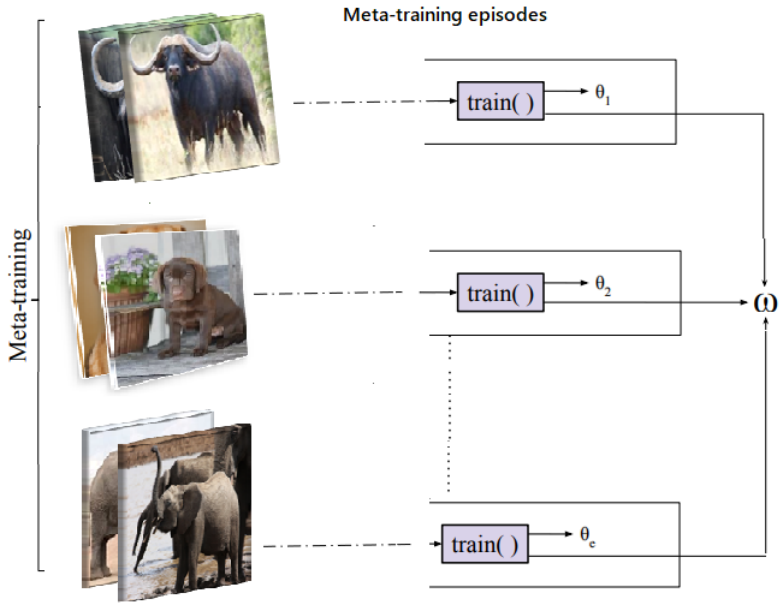


Figure 3.1: Meta-training episodes are characterised by a small number of samples representing novel classes with each episode having a new set of parameters θ_e $(e = 1...n \in \mathbb{Z})$ to be trained. During each training episode, the global deep learning network weights are affected by the new parameters changes, affecting the set of global parameters ω, and allowing knowledge to be accrued over episodes.

Meta-learning methods aim to acquire task-level meta-knowledge that can help the model quickly adapt to new tasks and environments with very few labelled examples [61]. A popular line of research, for instance, by [64, 137, 165, 204, 224] is to learn to fine-tune and aim to "obtain a good parameter initialization" [61, p. 2] that can adapt to new tasks with a few scholastic gradient descent updates. Some simple fine-tuning based approaches [29, 33, 46] turn out to produce better results than many prior works that use meta-learning on few-shot image classification. Another popular line of research for few-shot image classification is to use parameter generation during adaptation to novel tasks [32, 46, 71, 249]. Gidaris et al. [71] propose an attention-based weight generator to generate the classifier weights for the novel classes. Wang et al. [247] construct task-

aware feature embeddings by generating parameters for the feature layers.

The challenge in meta-learning is to learn from "prior experience in a systematic, data-driven" [165, p. 2] way. First, there is need to collect "meta-data that describe prior learning tasks and previously learned models, and to learn from this prior meta-data, to extract and transfer knowledge" [165, p. 2] that guides the search for optimal models for new tasks [64, 137, 165, 204, 224]. The more similar those previous tasks are, the more types of meta-data we can leverage [53, 87, 189], and defining task similarity will be a key overarching challenge. When a new task represents "completely unrelated phenomena, or random noise" [204, p. 1], leveraging prior experience will not be effective [87, 157, 189, 245]. There exists challenges in meta-learning that have remained largely unexplored. The use of single meta-learners heavily relies on trial and error hyperparameter tuning such as number of epochs, mini-batch sizes, and learning rates, to avoid over-fitting and under-fitting. The existing meta-learning methods do not consider the time and resource efficiency, making it difficult to meet real-world application requirements.

To tackle these challenges, we introduce a novel few-shot classification model that consists of dual meta-learners and a meta-ensemble module both supervised by a central controller (see Figure 3.3, and Figure 3.4) to control a feature extraction module and a meta-learning module, and a meta-ensemble module for integrated inference and generalisation. The ResNet-152 network with ImageNet pre-trained weights acts as a backbone that controls the feature extraction module, a meta-learning module, and meta-testing module for few-shot classification. Each meta-learner is composed of a pre-trained encoder fine-tuned by batch training and parameter-free decoder used for prediction. First, we train a feature extractor on all base categories to learn representations of inputs [31]. In the meta-training stage, the classifier is optimized in the metric space by "cosine distance" [286, p. 1] with a learnable scale parameter. Then in the meta-testing stage, the query sample in the unseen category is predicted by the adapted classifier given a few support samples.

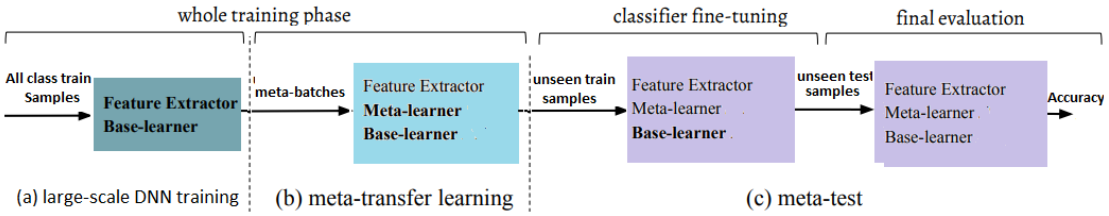


Figure 3.2: The pipeline of the meta-learning method, including: (a) training on the base dataset, (b) meta-transfer learning on the based on the parameters of pre-trained feature extractor. Learning is scheduled by the meta-learner; and (c) meta-test is done for a novel task which consists of a base-learner with fixed parameters. Then follows fine-tuning and a final evaluation stage.

Our model enables more effective initializations and faster adaptation, and has connections between instance-based information and semantic-based information. We consider the case of meta-learning based method that consists of mainly two stages: 1) meta-training, and 2) meta-testing. During the first meta-training stage, a sequence of episodes is randomly sampled from the labels of the base classes where each episode contains **K** support examples and **Q** query examples from **N** classes, denoted as an N-way K-shot episode (See Chapter 3). In this way, the meta-training stage can mimic the few-shot testing stage where only a few labels are available per class.

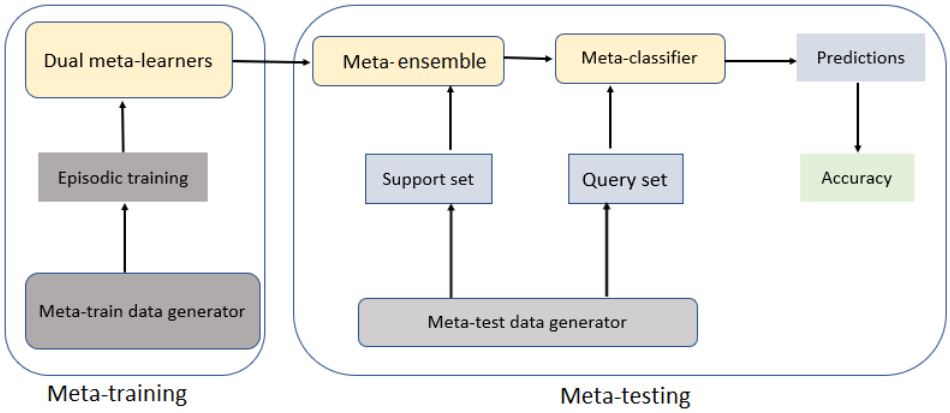


Figure 3.3: A meta-learner is first trained on the episodes of data generated by the meta data train generator. The meta-learner outputs a learner, which is then trained on the support set of each meta-test episode to output a predictor for evaluation.

The main contributions of this chapter are summarised as follows:

- We introduce a few-shot meta-learning model with dual learners supervised by a central controller that control a feature extraction and meta-learning, and a meta-ensemble module for integrated inference and generalisation. Each meta-learner is composed of a pre-trained encoder fine-tuned by batch training and parameter-free decoder used for prediction, first trained on a ResNet-152 backbone to learn image feature representations, with an optimize the classifier by using the cosine distance with a learnable scale parameter in the feature space in the meta-training stage.

- On the basis of empirical evaluation on the Omniglot, MiniImageNet, and Oxford Flowers102 datasets, we provide some insights for best practices in implementation.

The remainder of this chapter is organized as follows. In Section 3.3, we discuss the related work on meta-learners developed recently, and the state-of-the-art on few-shot classification approaches related to our work in this chapter. The proposed meta-learning

method is described in Section 3.4. We discuss the experimental results and illustrate the datasets used for training, and testing our model in Section 3.5. Some qualitative results are presented in Section 3.6. Finally, Section 3.7 concludes the chapter.

3.3 Related Work

The most successful methods that have been implemented in few-shot image classification belong to optimization-based methods [60, 165, 204], and metric-based methods [174, 214, 224, 239] (see Chapter 2). This has been achieved by learning a good parameter initialization for the classifier, and the learned weights can be quickly adapted to novel classes using gradient-based optimization on few labelled samples, or by learning a task-independent embedding, vector representations that can generalize to novel categories under a chosen distance metric such as a distance parametrised by a neural network, cosine distance, or Euclidean distance. The distance metrics provide a weighted nearest neighbour classifier representing each class with the average of the samples in the support set. Some recent works [30, 267] take advantage of both, and utilize meta-learning after pre-training, further boosting model performance.

Seminal work include MAML [60] that proposed a general optimization algorithm that poses the learning to learn problem in a bi-level optimization where the weights of the network are modelled as a function of the initial network weights. It aims to find a set of model parameters, such that a "small number of gradient steps with a small amount of training data" [60, p. 3] from a new task will produce large improvements on that classification task. Reptile [165] alleviates the expensive second order derivative computation in MAML by a first order approximation. It ignored the "second-order derivatives of MAML" [60, p. 3]. It achieved comparable results to complete MAML with orders of magnitude speed-up, and removed "re-initialization for each task, making it a more natural choice in certain settings" [60, p. 3]. MAML++ [6] introduces multiple speed and sta-

bility improvements over MAML. Meta-learner [190] exploited an LSTM to satisfy quick "acquisition of task-dependent knowledge and slow extraction of transferable knowledge" [190, p. 3]. LEO [204] proposed that it is beneficial to "decouple the optimization-based meta-learning algorithms from high-dimensional model parameters" [204, p. 1]. In particular, it learned a stochastic latent space from which the high-dimensional parameters can be generated. MetaOptNet [15] replaced the linear predictor with an SVM in the MAML framework; it incorporated a "differentiable quadratic programming (QP) solver to allow end-to-end learning" [15, p. 2]. Triantafillou et al. [234] showed that prototypical networks and MAML could be combined by leveraging prototypes for the initialization of the output weights value in the inner loop. MetaDelta [31] consists of multiple meta-learners, and is composed of a pre-trained encoder fine-tuned by batch training and parameter-free decoder used for prediction. It requires expensive computing resources to implement.

The work in this chapter is related to the rich literature on few-shot image classification which uses meta-learning, and metric-based methods for few-shot learning. This is the first work to conduct meta-learning analysis using dual meta-learners that are centrally controlled for feature extraction and meta-learning; and a meta-ensemble module for integrated inference and generalisation. Although our method is closely related to MAML [60] and MetaDelta [31], it benefits from data augmentation, transfer learning and metric learning. Similar to previous work [36, 190, 245, 278], our method also adopts a pre-trained ResNet-152 backbone to project images to latent vectors [6, 19], and adapts dual meta-learners to improve on the classification generalisation and improve on the time and memory resources. Contrary to MetaDelta, our method trains the classifiers in an episodic way on the training classes in few-shot settings. We implement a meta-ensemble module to improve the generalisation of the model's predictions.

3.4 Proposed Method

In this chapter, we propose a few-shot meta-learning [61, 165] model (illustrated in Figure 3.4) for few-shot learning that consists of a dual meta-learners with a central controller trained with different hyper-parameters, a feature extractor, a meta-learning stage and a meta-testing stage. The main process to start/stop the training process is centrally controlled. The two meta-learners are derived by training with different initial hyper-parameters also managed by the controller module. We leverage the optimised ResNet-152 CNN encoder trained on ImageNet to embed images into features, i.e. to map images to feature vectors, i.e. ResNet-152 weights have been re-used as the starting point for the training process and retrained to adapt to the new few-shot classification problem. The goal of meta-training is to minimize the N-way prediction loss in the query set. A base dataset D_{base} is used train the feature extractor that learns representations of inputs that will be used for further comparison in the feature space. All base categories are trained by minimizing a standard cross-entropy loss and removing its last fully-connected (FC) layer to get a 512-dimensional feature representation $f(\theta)$. We then add a classifier head onto the encoder for fine-tuning.

During the few-shot meta-training stage, a meta-learner classifier M is then trained over a set of episodes. The feature representation, $f(\theta)$, is treated here as an initial weight and optimised directly by minimising the generalisation error across episodes. We use the cosine distance to compare the feature representations for a single episode of training. During the meta-testing stage, we discard the classifier head and map the images to embeddings with the fine-tuned encoder, the classifier M is estimated on a set of episodes sampled from the novel meta-test set D_{novel}. Then, we can compare the query images with the images in the support set by comparing the embeddings' pairwise cosine similarities, and use similarity scores to make predictions.

3.4.1 Dual meta-learners

Two different meta-learners (see Figure 3.4) are derived by training with different hyper-parameters in parallel for resource efficacy. They are both controlled by a central controller that dispatches the meta-data, and also decides when to start and stop the meta-training and testing. We apply episodic training to learn a CNN encoder for feature vector embedding for many epochs r. Then a decoder is used during the meta-validation and testing periods to decode the vectors of each episode to the predicted labels for the classification accuracy.

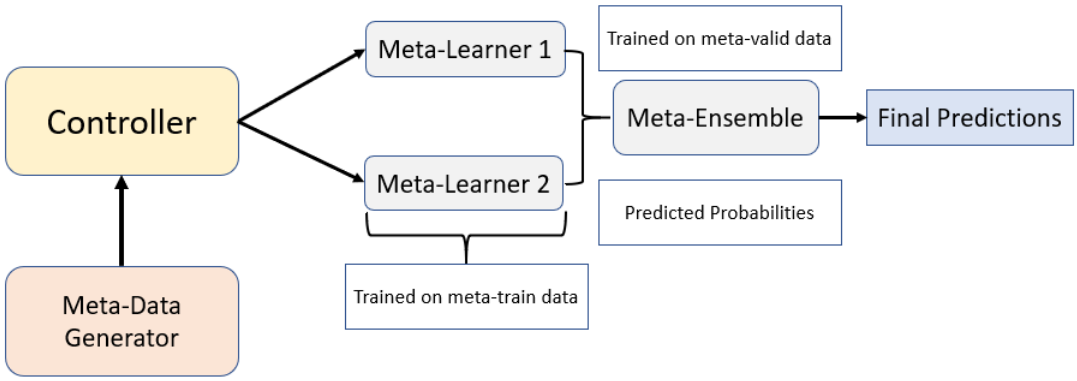


Figure 3.4: Dual meta-learner and a meta-ensemble module to improve generalisation of the model.

3.4.2 Meta-ensemble

The meta-ensemble module (see Figure 3.4) integrates the predicted probabilities of the two meta-learners and outputs the final predictions. This design further improves the robustness of our system. The meta-ensemble module is trained after finishing the meta-training of all meta-learners. To train the meta-ensemble model, we divide the meta-valid data into a training set and a test set. Taking the concatenation of the predicted probabilities from the two meta-learners as input, several meta-ensemble models are trained on the training set simultaneously and evaluated on training set based on episodic accuracy.

The best meta-ensemble model is then saved for the inference in the meta-test period. In our experiments, we implement Naive Bayesian Classifier, Gradient Boost Machine, Random Forest, and General Linear Model as implemented in Scikit-Learn [177] as the meta-ensemble candidate models. Due to the diversity of suitable scenarios of these models, we argue that our meta-ensemble module is capable of dynamically adapting to the unknown feedback dataset by selecting the best ensemble model according to the meta-valid data.

3.4.3 Feature extractor

We select a pre-trained backbones, fine-tune, and train a feature extractor f_θ with parameters θ on the base set D_{base} that encodes the input data to a 512-dimensional feature vector suitable for comparison. Here, we employ ResNet-152 backbone to learn a classifier on all base categories and remove the last fully connected layer to get $f(\theta)$. Before feeding to the network, all input images in D_{base} are resized to 80×80. The architectural setting of ResNet-152 (see Figure 3.6).

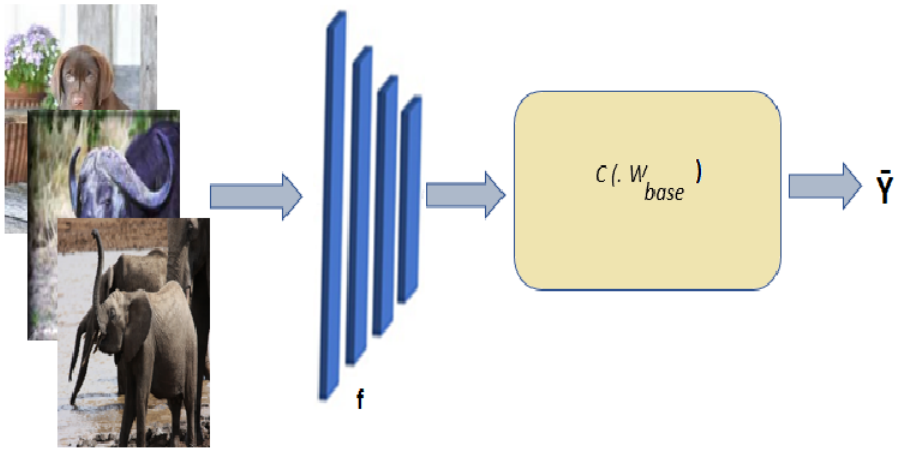


Figure 3.5: The feature extractor trained on the base dataset by removing the fully-connected layers, and generating the feature encoder f_θ

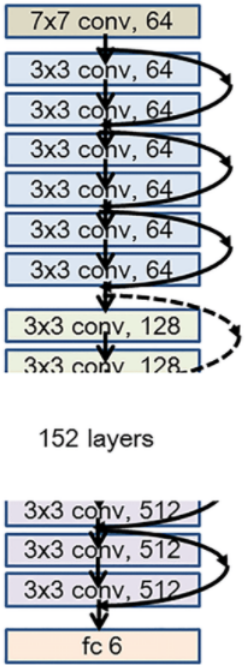


Figure 3.6: ResNet-152 Model.

3.4.4 Meta-training stage

We adopt the N-way K-shot setting few-shot meta-training [203] to extract meta-knowledge from the set of episodes. The goal is to train a meta-learning model $\mathcal{M}(.|S)$ that minimizes the N-way prediction loss. This is accomplished by sampling many episodes from the meta-training data in base categories. Each one of the episodes has K input samples and the same number of output samples. The parameters of classifier M are shared across all the episodes resulting in reduced requirement for large samples during model training. The samples are randomly selected from each category, i.e., a total of $N \times K$ samples for N-way classification training and $N \times Q$ query samples for meta-testing. A meta-validation set is held out for the purposed of choosing the hyper-parameters of the model $\mathcal{M}(.|S)$ during meta-training. Figure 3.7 illustrates the workflow of the proposed meta-training stage. With each episode with the support-set $\mathcal{S}$, we denote S_c as a subset of S with all samples in category c defined a prototype $\mathfrak{w}_c$ as the mean vector over embed-

dings, or representations belonging to S_c (the centroid of category c). An embedding is generated by the pre-trained feature extractor f_θ with learnable parameters θ we described in 3.4.3. We write down the $\mathfrak{w}_c$ as follows:

$$\mathfrak{w}_c = \frac{1}{|\mathcal{S}_c|} \sum_{x_i \in \mathcal{S}_c} f_\theta(x_i) \tag{3.1}$$

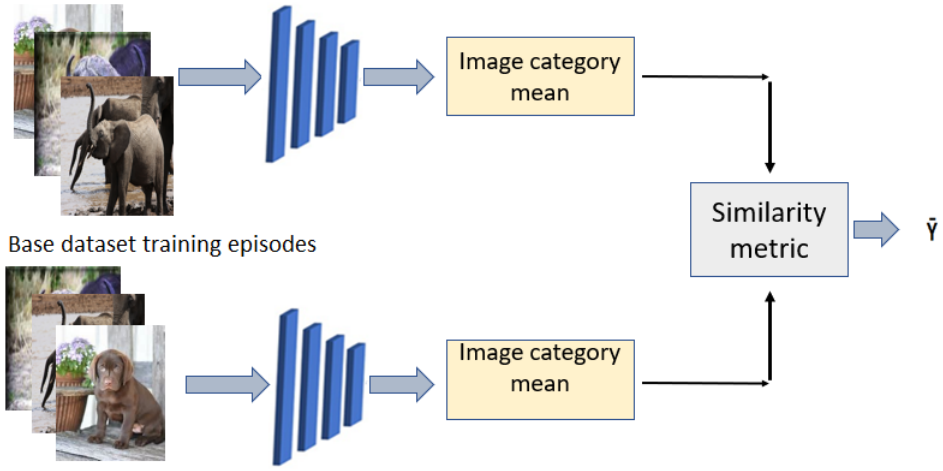


Figure 3.7: Proposed meta-training stage for a N-way K-shot classification.

We predict the probability that a query sample x belongs to category c. We compare the distance between the feature embedding $f_\theta(x)$ and the centroid $\mathfrak{w}_c$ of category c, using the cosine similarity, and thus the prediction can be formalized as follows:

$$p(y = c/x) = \frac{\exp(\cos(f_\theta(x), \omega_c))}{\sum_{c^i} \exp(\cos(f_\theta(x), \omega'_c))} \tag{3.2}$$

Inspired by metric learning few-shot learning methods [91, 105, 110, 172] , we introduce α, a learnable scalar parameter to adjust the original value range $[-1, 1]$ of cosine similarity. In our experiments, α is initialized to 10 following the work by [31]. We

observe that the scaling similarity metric is more appropriate for the following softmax layer. Thus, the predictive probability becomes:

$$p(y = c|x) = \frac{\exp(\alpha \cdot cos(f_\theta(x), \omega_c))}{\sum_{c^i} exp(\alpha \cdot cos(f_\theta(x), \omega_c^{'}))} \tag{3.3}$$

3.4.5 Meta-testing stage

After meta-training, and the meta-learning model $\mathcal{M}(\cdot|S_{base})$ is learned, we evaluate its generalization ability on a held-out novel set D_{novel} that has been unseen during the meta-training stage. At this stage, we are given new episodes sampled from D_{novel}, often referred to as a meta-test set $D_{\mathcal{T}}^{test}$, $\mathcal{T} = \{(\mathcal{S}_{novel}, \mathcal{Q}_{novel})\}$. The learned model is therefore, at this stage adapted to predict novel categories with the new support set $\mathcal{S}_{novel}$.

3.5 Experimental Results

We present some implementation details and dataset description in this section. Then, we compare our method with some state-of-the-art few-shot classification methods. Furthermore, we carry out experiments on three public benchmark datasets to demonstrate the effectiveness of the proposed method for few-shot object classification. In this work, we also selected support and query images of ten animals of each of the antelope, the sloth, the moose, the jackal, the squirrel, the hedgehog, the penguin, the wild dog, the kori bustard, and the meerkat from the Internet. The idea was to include animals that are not well-represented in the MiniImageNet dataset, and use these for few-shot training to gauge the performance of our model.

3.5.1 Datasets

We evaluate our proposed method on three public datasets, Omniglot [122], MiniImageNet [239] and Oxford Flowers102 [166]. For the Oxford Flowers102 dataset, we randomly partitioned the dataset classes into meta-training, meta-validation, and meta-testing according to the approximate ratio of 7:1:2 (see Table 3.1). Each of the three datasets is divided into three sets: meta-train-support, meta-train-validation held-out for hyper-parameter selection of the meta-training stage, and meta-train-query. The set split for meta-training are the 7 out of 10 categories of D_{base}. For instance, the Oxford Flowers102 dataset with overall 102 categories of flowers in the United Kingdom has approximately 71 categories for meta-training, 21 categories for meta-validation; and 10 classes for meta-testing. For the Oxford Flowers102 dataset, the number of images in each category is shown in Table 3.2.

Omniglot is a dataset of "hand-written characters with 1623 characters and 20 examples of size 150×150 " [122, p. 1] for each character, collected based upon 50 alphabets from different countries. The miniImageNet dataset contains 100 classes randomly chosen from ImageNet ILSVRC-2012 challenge with 600 images of size 84×84 pixels per class. It is split into "64 base classes, 16 validation classes and 20 novel classes" [239, p. 1]. The Omniglot and MiniImageNet are already designed for few-shot learning. All the three datasets are publicly available. We follow the same split for the MiniImageNet dataset; the only change is the number of images in each category. For our purposes, we select only up to 10 training images from each category depending on whether we are testing for 1-shot, 5-shot or 10-shot.

Table 3.1: Approximate datasets split

Dataset	Meta-train	Meta-validation	Meta-testing
Omniglot	35	5	10
MiniImageNet	64	20	16
Oxford Flowers102	71	21	10

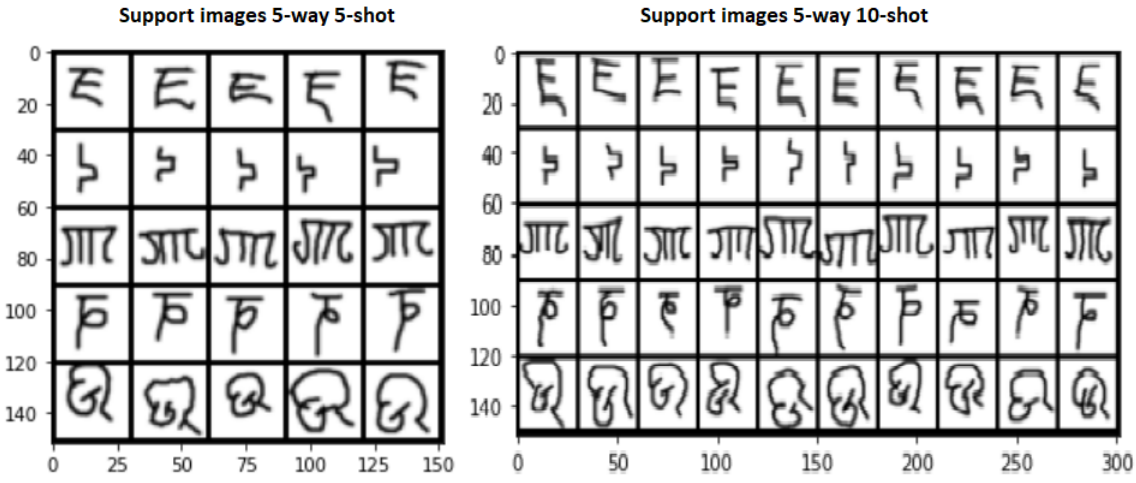


Figure 3.8: Omniglot 5-way 5-shot, and 5-way 10-shot tasks.

3.5.2 Implementation details

We follow the **N**-way classification with **K**-shots [239] few-shot experimental protocol. In other words, we create a data loader that evenly distributes the images (see Figure 3.8 in case of Omniglot) between the given number of classes from each of the Omniglot, Oxford Flowers102 and the MiniImageNet datasets, and always split them between support and query sets before feeding the few-shot classification tasks. The Omniglot and MiniImageNet have already been split. The **N**-way dataset classes is sampled into **N**-shot and **N**-query images for each class in each batch. Each batch which is fed into the task is

Table 3.2: Oxford Flowers102 dataset split

Dataset Split		**Number of Categories**	**Images per Category**
base	train-support	102	1020
	train-validation	102	204
	train-query	102	102
validation	meta-validation	102	102
novel	meta-test	102	204

a combination of the support images, support labels between **0** and **N**-way, query images (also between **0** and **N**-way), and a mapping of each label between **0** and **N**-way to its class label in the dataset.

Here **N** = 5, **K** = 10, i.e., the meta-training stage consists of several episodes, with each episode being a selection of 5 randomly categories drawn from D_{base} in the meta-training stage. Following episodic training, we set 4 episodes per batch to compute the average loss for the batch size of 4. The support set in each training episode is expected to match the same number of shots as in the meta-test stage. During the meta-testing stage, we perform 5-way 1-shot classification at meta-testing time, then the training episodes could be constituted of **N** = 5, **K** = 1. Each category contains **K** query samples with 15 query samples. There is a limited number of samples for meta-training in each episode, but the number of episodes is large enough, with an epoch that contains up to 1000 episodes depending on the dataset size and steps per epoch. We ensure that there are at least 50 epochs for training. We can therefore, assume that the entire datasets have largely been traversed.

3.5.3 Baselines

Some representations of optimisation-based and metric-based meta-learning methods are adopted as our baselines. These two have provided some of the best results in few-shot classification. MAML [60] and its enhancements transforms the inner-update gradients to improve generalisation capacity. ProtoNet [214] is chosen because it applies episodic training rather than batch-training. RelatioNet [224] is a relational network that learns a distance metric to compare images within episodes in few-shot settings, while TADAM [172] utilises a task-adaptive metric. Our model adopts ResNet-152 that is pre-trained on the ImageNet dataset. We have re-implemented their versions with a ResNet-152 backbone for a fair comparisons. ProtoNet, RelationNet and TADAM are metric-based methods. ProtoNet uses Euclidean distance while RelationNet compares an embedding f_φ and query samples using an additional parametrised CNN-based methods. TADAM assumes a task-conditioned feature extractor should be more discriminative for a given task. They presented a dynamic feature extractor that can be optimized by a given support set S.

3.5.4 Results and comparison

We conduct experiments to evaluate the effectiveness of our method following the 5-way 10-shot, and 5-way 5-shot, and 5-way 1-shot (see Figure 3.13). The proposed method is compared with various state-of-the-art few-shot learning methods. For 5-way 10-shot experiment, ten labelled support samples per category is randomly selected as the supervised sample at the test time. For 5-way 1-shot experiment, one labelled support sample per category is randomly selected as the supervised sample at the test time. Likewise, 5 support samples per category are provided for 5-shot setting. Query images are selected according to individual categories, and are batched in each episode for evaluation. We computed the mean classification accuracy of the randomly generated episodes from the novel meta-test set for each **N**-way **K**-shot.

RelationNet, and ProtoNet were all originally implemented on a Conv-4. We decided to re-implement them with a ResNet-152 backbone for a fair comparison. For MAML, we adopt the first-order version for the experiments re-implemented with ResNet-152. The original paper reported identical results to the version with second order derivatives. On the three datasets, the results of average 5-way accuracy (%) with 95% confidence interval of 1-shot and 5-shot are reported in Tables 3.3, 3.4, and 3.5 respectively. As can be seen, our method performs comparatively with the other models under both 5-way 10-shot and 5-way 5-shot settings.

The figures below show training and validation metrics for training the three datasets that were used for the experiments. Our model achieves 69.5% top-1 accuracy on Omniglot, 62.4% on Oxford Flowers102, and 60.3% on the MiniImageNet, and a top-5 accuracy of 73.72% on Omniglot, 63.38% on Oxford Flowers102, and 65.72% on the MiniImageNet. The model consistently achieves higher accuracy levels with 10 shots. With more epochs of training, it can achieve a better performance.

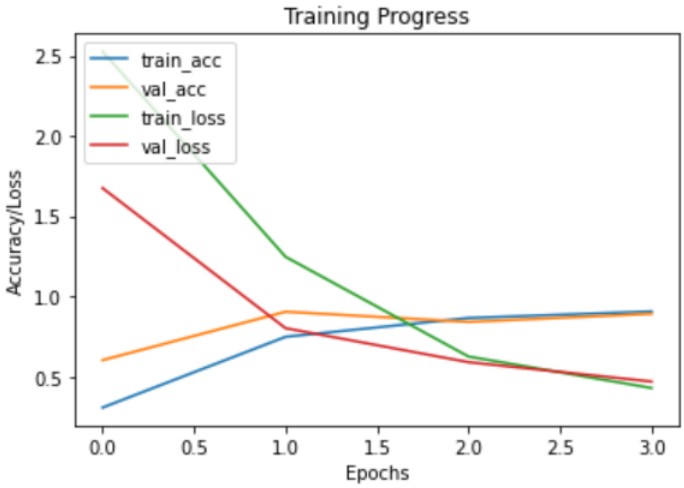


Figure 3.9: Few-shot training progress for the Omniglot dataset.

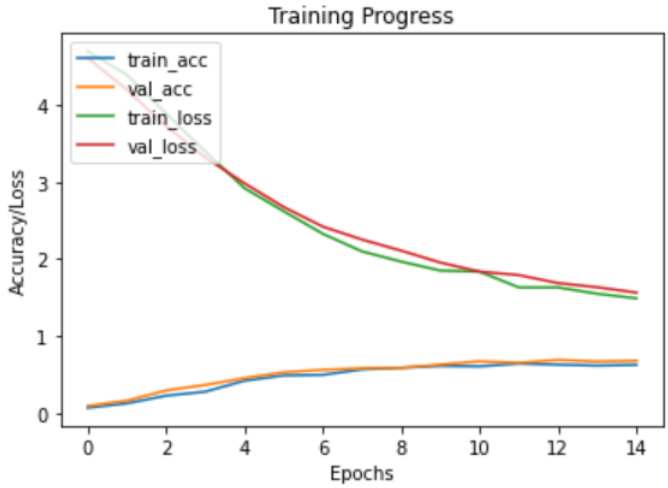


Figure 3.10: Few-shot training progress for the Oxford Flowers102 dataset.

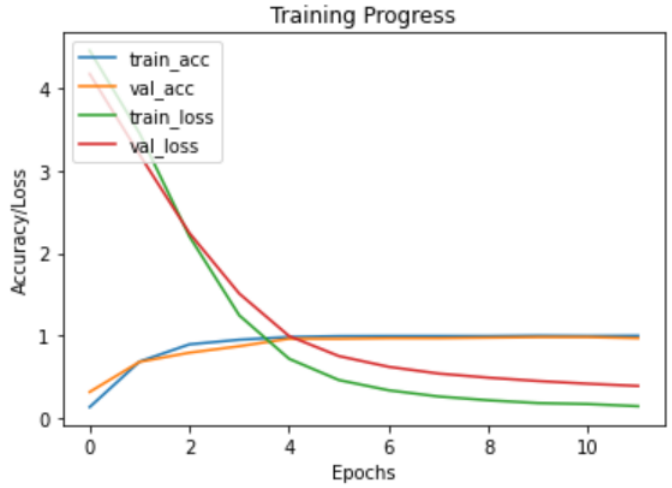


Figure 3.11: Few-shot training progress for the MiniImageNet dataset.

We also evaluate our model using a completely new, simple animal image dataset downloaded from the Internet consisting of 100 images divided into 10 animals of each type (see Figure 3.12). The results were normalised with a $threshold > 0.5$. The Confusion Matrix in Figure 3.12 show the results after running the tests. After testing the model with 10 of each animals after few-shot classification training, between 50% and

80% of each of the animals was correctly classified, indicating that the model effectively separated among the different representations of each of the animals.

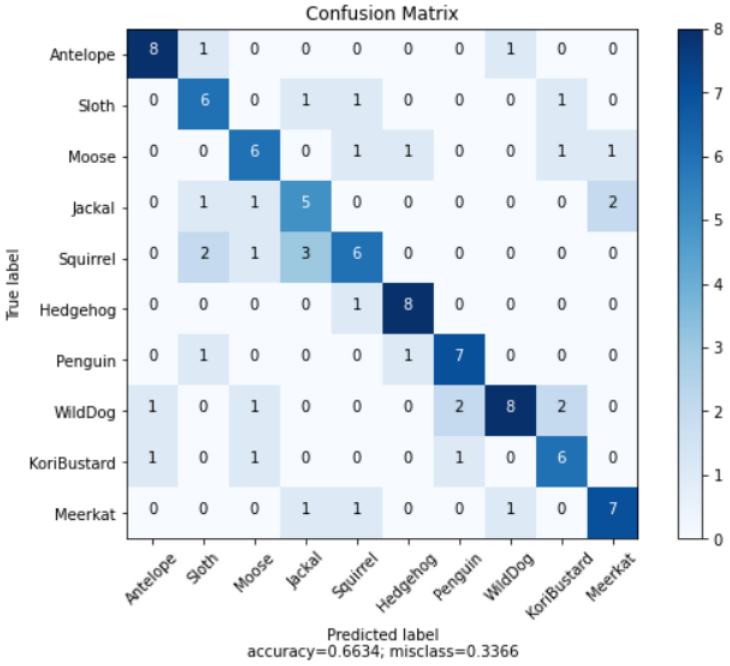


Figure 3.12: The Confusion Matrix for multi-class (10) few-shot classification after training on the MiniImageNet base dataset and 10-shot classification on few examples. The selected 10 classes used are 'Antelope', 'Sloth', 'Moose', 'Jackal', 'Squirrel', 'Hedgehog', 'Penguin', 'WildDog', 'KoriBustard', and 'Meerkat'.

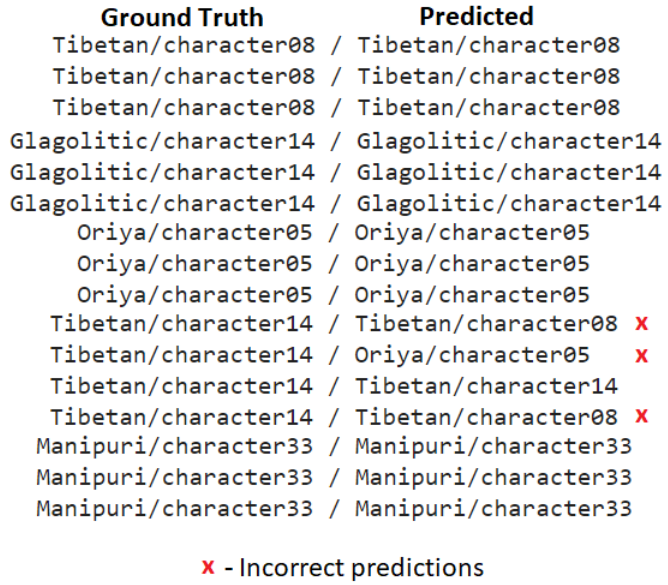


Figure 3.13: Omniglot 5-way 5-shot tasks example predictions.

The Tables 3.3, 3.4, and 3.5 show the reported averaged results of the selected baselines, and those of our re-implementation (with a *) of MAML, ProtoNet, and RelationNet with ResNet-152 backbone. We observed that a deeper backbone such as Resnet-152 slightly improves MAML and RelationNet. Moreover, all the implementations seem to get better in the 10-shot case for the three datasets when implemented with ResNet-152. Typically, CNN-based methods most likely lead to over-fitting when there are only a few labelled examples contrary to what has been achieved by meta-learning methods. On the other hand, ProtoNet is improved by a large margin when the backbone architecture is replaced with Resnet-152, which shows that ProtoNet is a powerful and robust approach. Basing on the results, we conclude that our model compares favourably against the selected methods in both shallow and deeper backbone settings.

Table 3.3: Comparisons of few-shot classification on Omniglot using various models and backbones, and our model. Bold indicates the best model for each shot category. ResNet backbones load the pre-trained weights from ImageNet. * indicates re-implementation with ResNet-152.

Method	**Backbone**	**1-Shot**	**5-Shot**	**10-Shot**
ProtoNet [214]	Conv4	45.6	77.65	94.58
ProtoNet* [214]	ResNet-152	70.8	79.17	**97.67**
MAML [60]	Conv4	56.8	73.52	89.48
MAML* [60]	ResNet-152	73.6	**83.52**	96.74
RelationNet [224]	Conv4	56.7	72.10	67.67
RelationNet* [224]	Resnet-152	65.9	76.56	93.55
TADAM [172]	Conv4	54.4	72.25	97.46
TADAM* [172]	Resnet-152	**74.3**	82.72	96.41
Ours	Resnet-152	64.1	73.72	97.52

3.6 Qualitative Results

We present some qualitative results on Omniglot, MiniImageNet and Oxford Flowers102 datasets in Figure 3.14 to Figure 3.20. Most of the images were correctly classified with a probability close to 1. The classification percentage was highest for the Omniglot dataset, followed by the Oxford Flowers 102 and the ImageNet datasets. For the dataset of images downloaded from the internet, the probabilities indicated for the animals on the image were high, indicating that our model correctly classified the images in most instances, and this confirms our results on the Confusion Matrix (see Figure 3.12). There are a few other animal images that were misclassified, for instance, one moose image on Figure 3.19 was misclassified as most likely to be a warthog.

Table 3.4: Comparisons of few-shot classification on MiniImageNet using various models and backbones, and our model. Bold indicates the best model for each shot category. ResNet backbones load the pre-trained weights from ImageNet. * indicates re-implementation with ResNet-152.

Method	**Backbone**	**1-Shot**	**5-Shot**	**10-Shot**
ProtoNet	Conv4	43.1	52.57	71.95
ProtoNet*	Resnet-152	52.57	64.3	81.58
MAML	Conv4	52.73	55.6	69.28
MAML*	Resnet-152	50.9	57.76	79.81
RelationNet	Conv4	46.4	53.73	78.86
RelationNet*	Resnet-152	54.5	63.70	**88.86**
TADAM	Conv4	50.4	65.84	82.79
TADAM*	Resnet-152	**59.5**	**72.05**	87.60
Ours	Resnet-152	54.6	65.72	81.62

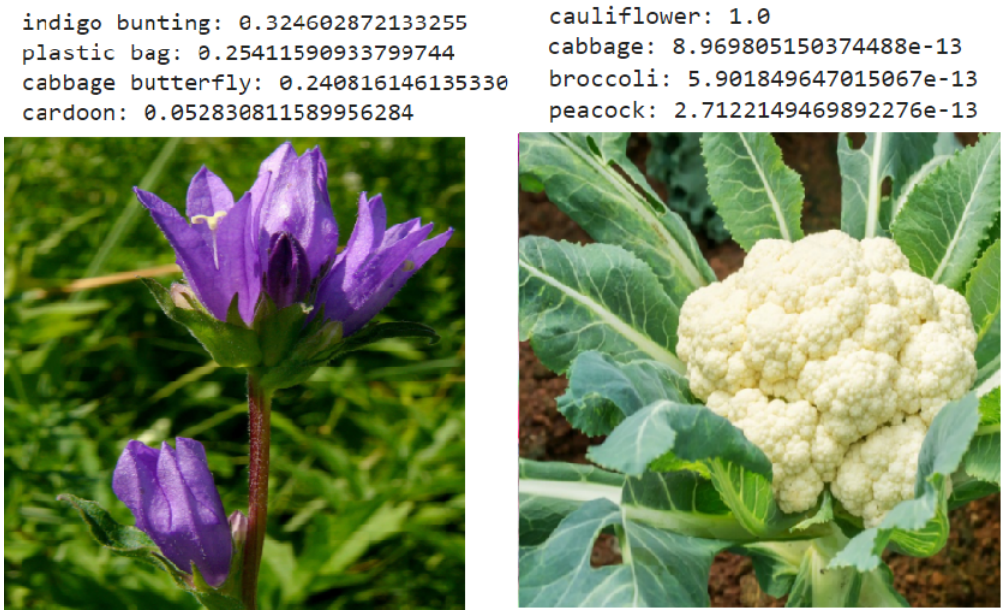


Figure 3.14: Predicted result on few-shot classification on flower images.

Table 3.5: Comparisons of few-shot classification on Oxford Flowers102 dataset using various models and backbones, and our model. Bold indicates the best model for each shot category. ResNet backbones load the pre-trained weights from ImageNet. * indicates re-implementation with ResNet-152.

Method	**Backbone**	**1-Shot**	**5-Shot**	**10-Shot**
ProtoNet	Conv4	51.65	69.58	87.4
ProtoNet*	ResNet-152	57.17	79.18	91.8
MAML	Conv4	53.52	70.94	88.5
MAML*	ResNet-152	57.52	77.94	93.3
RelationNet	Conv4	58.10	72.55	87.7
RelationNet*	Resnet-152	53.10	79.87	93.6
TADAM	Conv4	62.25	79.36	86.7
TADAM*	Resnet-152	**66.72**	**84.41**	**95.2**
Ours	Resnet-152	63.68	79.52	93.4

Figure 3.15: Predicted result on few-shot classification on beetle and coyote images.

Figure 3.16: Predicted result on few-shot classification on mushroom images.

Figure 3.17: Predicted result on few-shot classification on a penguin and kori bustard image.

Figure 3.18: Predicted result on few-shot classification on a sloth and the squirrel image.

Figure 3.19: Predicted result on few-shot classification on a wild dog and moose image.

Figure 3.20: Predicted result on few-shot classification on a meerkat and a hedgehog image.

3.6.1 Ablation studies

We further conduct some ablation studies to demonstrate the functionality of backbones and decoders. We implement single meta-learners with ResNet-50, ResNet-101, and MobileNet, and applied decoders of ProtoNet.

Table 3.6: Comparisons of few-shot classification using different backbones. Bold indicates highest among the backbones for each selected dataset.

Backbone	**Omniglot**	**MiniImageNet**	**Oxford Flowers102**
ResNet-50	97.54	**78.67**	74.58
ResNet-101	**98.63**	77.76	76.23
MobileNet V2	95.78	76.04	**78.94**

We also investigate the impact of the use of either the Euclidean distance or cosine similarity distance metrics for the few-shot classification. The two choices are compared in (see Table 3.7). As shown in Table 3.7, the performance improves to 81.86% and 79.52%, respectively, in the 5-way 10-shot case, a gain of more than 5 % each.

Table 3.7: Comparisons of 5-shot few-shot classification accuracies between Euclidean distance and cosine similarity on MiniImageNet and Oxford Flowers102

Metric	**Omniglot**	**MiniImageNet**	**Oxford Flowers102**
Euclidean	67.01	60.3	65
Cosine	73.72	65.72	79.52

3.7 Conclusion

The meta-learning framework in few-shot learning has attracted much attention in recent years. In this chapter, we bring dual meta-learners and a meta-ensemble module to meta-learning, and demonstrate that useful information may be learnt from a few image instances. The proposed model generalises well to unseen categories after training on a few samples. We have employed a pre-trained ResNet-152 to learn vector representations on the base set, and the classifier is optimised by the cosine distance. Our experiments, conducted on the Omniglot, MiniImageNet, and Oxford Flowers102 datasets, and few, selected images from the Internet achieve a general classification performance above 70% for a novel categories on 5-shot, and more than 80% classification accuracies for 10-shot settings. Furthermore, we conducted some ablation experiments to investigate the effects of different other network backbones, and the impact of the use of different distance functions with dual meta-learners. We conclude that the cosine distance generally has a better performance.

In the next chapter a novel approach for few-shot object detection that meta-learns object localisation and instance categorisation is proposed. In line with few-shot learning settings, and using the Transformer decoder-encoder architecture, support set images and query set images are simultaneously encoded into class-specific features that are input into a class-agnostic decoder to generate predictions for the specific instances. A module is designed that aligns semantics of high-level and low-level features representations, and all the modules are designed in multi-scale end-to-end architecture.

Chapter 4

Few-shot Object Detection through Image Object Localisation using The Transformer

The previous chapter introduced a novel meta-learning model for few-shot image classification that consists of dual meta-learners supervised by a central controller for integrated inference and generalisation with more effective initialisations and adaptations to novel data using a pre-trained feature encoder. In this chapter, we propose an approach for few-shot object detection that, instead of region-wise predictions, meta-learns "object localisation and classification" [278, p. 1] in a "end-to-end" [142, p. 1] manner, and encodes input images into feature embeddings that are entered into a class-agnostic decoder to output predictions for the identified object classes.

4.1 Introduction

Object detection (see Chapter 2) has been a long-standing problem in computer vision fields. It generally deals with identifying the location of target objects in the input image

as well as recognizing the object categories, with many real-world applications, such as in remote sensing, change detection, environmental monitoring, and urban planning. A large number of methods have been developed for the detection of both artificial objects such as buildings, vehicles, and airports; and natural objects such as shorelines, lakes, and forests in remote sensing images. Among the object detection methods, object-based image analysis (OBIA)–based methods and deep learning–based methods have recently been used. The deep learning-based methods, especially convolutional neural networks (CNNs) have powerful abilities for robust feature extraction and object classification and are extensively studied by many recent approaches [74, 75, 146, 191, 194, 235].

Remarkable achievements in object detection from images have previously been reported with CNN-based methods such as Region-Based Convolutional Neural Networks (R-CNN) [74], "You only Look Once (YOLO)" [194, p. 1] and "Single Shot Detectors (SSD)" [146, p. 1]. Despite the breakthrough achieved by these methods, they require a large-scale, diverse annotated datasets in supervised settings to successfully train a deep neural network learning model. Any adjustment on the candidate identifiable classes will be expensive for existing methods because collecting a new dataset with a large number of manual annotations is costly [239]. Additionally, these methods require a lot of time to re-train their parameters on the new unseen dataset. They also tend to have limited generalisation abilities for unseen object categories. On the other hand, training these models with only a few samples from the new classes tend to suffer from the over-fitting problem and the generalization errors [213]. Therefore, a novel methods of learning and/or selecting the most robust and desirable features from a few samples during training is desired for object detection.

The goal for few-shot object detection is to detect objects from images given a few support images of novel target object. A pre-trained object detection backbone model is usually used first for general feature extraction. Few-shot learning is a challenging problem given large variance of objects' illumination, shape, and texture in images. Central to

few-shot object detection is how to localize an unseen object in a background given only a few image samples. Also, there could be more than one object instance on the image, and the object detection systems have to locate all of them, and say what they are, a more difficult task than in image classification where the task is just to identify the class of the image.

In this chapter, we address the problem of few-shot object detection (see Figure 4.1). We introduce a novel approach for few-shot object detection. Our approach uses meta-learning specifically for object localisation and categorisation in an end-to-end manner by eliminating region-wise prediction. Our method, based on DETR [20], encodes input images into feature embeddings that are finally input into a class-agnostic decoder to produce class predictions for the identified object categories. To facilitate meta-learning, a module is designed that aligns semantics of "high-level and low-level features" [127, p. 3] representations. All the modules are designed in "multi-scale architecture to enable multi-scale object detection" [44, p. 3].

Figure 4.1: Illustration of object detection. For few-shot object detection, the training images are evenly distributed among a given number of classes, and they are split between annotated samples of the support set and query set.

We can state the main contributions of this chapter as:

- We introduce a novel approach for object detection in few-shot settings that implements meta-learning for object localisation and categorisation by eliminating region-wise prediction. Support set images and query set images are simultaneously encoded into class-specific features that is subsequently entered into a class-agnostic decoder to output class predictions for the identified classes of images. To facilitate meta-learning, a module is designed that aligns "semantics of high-level and low-level features" [127, p. 3] representations. All the modules are designed in "multi-scale architecture to enable multi-scale object detection" [44, p. 3].

- Experiments on two public benchmark datasets, COCO-20^i and Pascal 5^i demonstrate the effectiveness of the proposed method for "few-shot object detection" [55, p. 1] on a variety of images.

Our method pursues meta-learning approaches. Unlike prior work that has been done in few-shot learning, the model discards region-wise prediction. It works by unifying the learning of object localisation and categorisation at image level with a image class-agnostic decoder, thereby combining the relationship of the two sub-tasks to achieve object detection performance.

The remainder of this chapter is organized as follows. In Section 4.2, we discuss the related work on the various state-of-the-art few-shot object detection [55] approaches that were developed recently, and are related to our method. The proposed object detection meta-learning based method is described in Section 4.3. We illustrate the datasets used for the experimental results in this chapter and discuss experimental results in Section 4.4. Finally, Section 4.6 concludes the chapter with a summary and an outlook.

4.2 Related Work

Two paradigms that have primarily been used to formulate models on few-shot object detection [55] are transfer-learning-based, e.g. LSTD [24], TFA [248], MPSR [256], whereby new concepts are acquired via fine-tuning; and meta-learning-based methods, e.g. Meta-YOLO [101], Meta R-CNN [263], ONCE [178], and FSOD [55], that acquire meta-level representations to adjust to new classes of objects by learning on other auxiliary tasks. The output instance classes are programmatically controlled on the support set from the dataset. Meta-YOLO, for instance, and ONCE are based on single-stage object detectors. Others, such as Meta R-CNN and its various variations such as [136, 256, 258] are built upon Faster R-CNN [84]. Their only limitation is that well-located regions for new instance objects are generally difficult to get with shape priors that are non-learnable when training samples are scarce, and where there is need for "per-region classification and location fine-tuning. Attempts to limit the effects with this issue by meta-learning an Attention-RPN in FSOD which is innately region-based has not solved the problem.

Our work in this chapter is related to meta-learning methods [55, 178, 248, 258, 259, 278] that have proved promising to few-shot object detection [55]. These generally address object detection by meta-learning over image regions. They include anchors [101] and region proposals [258, 259] for object identification, classification and fine-tuning. Most existing meta-detectors depend entirely on the initial region proposal for region-wise predictions. This cannot be guaranteed in few-shot learning settings. This chapter presents a framework that meta-learns image-level localisation in an end-to-end manner, facilitated by the emergence of end-to-end transformer frameworks [20, 139, 243, 278, 293]. Our work incorporates meta-learning into DEtection TRansformer (DETR) [20, 294] by encoding input images into feature embeddings, and feeding them into a decoder for detecting target object categories. DETR is based on the architecture of Transformer [237] and assign a unique query for each ground truth through bipartite matching. The idea of using a transformer here is that it can use its self-attention mechanism to per-

form global reasoning on the image as well as on the specific objects that are predicted, and also get rid of duplicate predictions. The model gets rid of region-wise prediction and merges the meta-learning of localisation and categorisation at image level with a class-agnostic decoder. All objects are predicted in parallel rather than having a sequential prediction. The model may look at other regions of the image to help make a decision about the object in a bounding box, and makes predictions based on relationships between objects in an image. In contrast, other detection models such as Faster R-CNN predict each object in isolation.

4.3 Proposed Method

The problem of few-shot object detection aims at learning a detection model from the base classes with adequate samples for model training that can conduct object detection on images from novel classes with only a few annotated samples. Our method for few-shot object detection extends DETR by incorporating meta-learning. In contrast to Faster R-CNN [198], DETR [20] achieves fully end-to-end detection by employing a "transformer encoder-decoder architecture" [237, p. 1] merged with a loss founded on the Hungarian algorithm [118] that strengthened by distinctive output predictions for each instance image object via "bipartite matching" [20, p. 1]. In this work, we use episodic training for meta-training.

4.3.1 Method overview

Figure 4.2 illustrates the pipeline of the proposed method. Our method is designed to leverage the meta-knowledge from the dataset of base classes. To achieve this goal, it implements meta-learning in DETR [20] built upon the transformer encoder-decoder architecture. The method consists of a shared transformer encoder for the support and query images, and a transformer decoder to output the class prediction of objects of the the in-

put support set classes. Support and the query encoders receive the query and support images respectively from a dataset, and extract support and query feature image embeddings. The transformer decoder first sums them into image instance class-specific feature embeddings. An instance class-agnostic transformer decoder is then applied (see Figure 4.2.

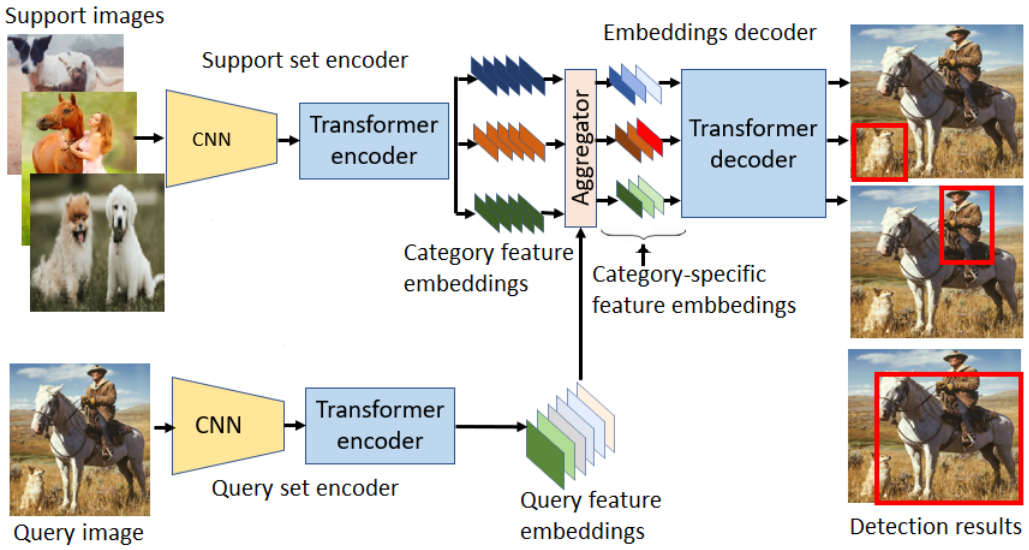


Figure 4.2: The design of the proposed method. It is composed of a shared transformer encoder for receiving the query and support images respectively and extract support and query feature image embeddings through the image feature extractor and the image transformer encoder, and a transformer decoder that first aggregates the images into class-specific feature embeddings. It then applies a transformer decoder for output of the results.

4.3.2 Model description

Our model consists of an encoder-decoder architecture that supports the input of annotated support images and a query image. The encoder for support images and that for the query image share all the learnable parameters following the architecture of the Siamese

networks [113]. The encoder consists of a feature extractor based on ResNet-101 [83] and a transformer encoder. The feature extractor generates its feature maps and then adopts a 1×1 convolution for compatibility with downstream modules. The feature maps are then fed into a transformer encoder to output the query features. The support and the query encoder encodes them into category feature codes and query feature codes respectively. The support encoder, therefore, extract class instance codes belonging to certain object instances, and to filter out irrelevant information, including in the case of many support images belonging to one class. The decoder takes input of the image query features and the class feature codes, and predicts the results over the corresponding support classes. It sums up, following previous work by [258], the query features and category codes into a set of class-specific features (see Figure 4.3). A transformer decoder then takes the features, and a fixed number of object queries and produces detection results over the corresponding categories which enables joint meta-training of object localisation and classification. This, therefore, eliminates region-wise prediction and address object detection at image level, also being category-agnostic like in DETR.

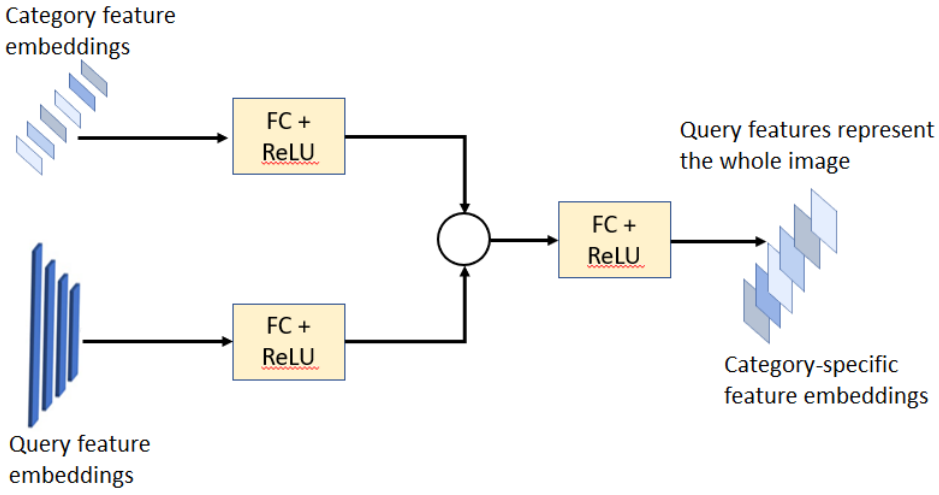


Figure 4.3: Illustration of aggregation between category query codes and the positions of query features in the decoder.

To mitigate against relying on category-specific semantics for object detection, we adopt the "Semantic Alignment Mechanism" (see [278, p. 4]) as a self-regularisation mechanism. This will assists as a guiding template for the feature semantics from the transformer encoder to orient with the fed input feature semantics with better generalisation by using a residual connection that bypasses the entire transformer encoder.

4.3.3 Training

The training procedure consists of two stages, 1) base training stage, and 2) few-shot fine-tuning stage. During base training, the model is trained on the base dataset D_{base} with large quantities of support set and query set samples for each base class. During the second stage, it is trained the base and novel class categories simultaneously but with a few training image data. Following [258, 263], we include several object instances for each base category. The network is optimised in an end-to-end manner with episodic training, with each episode containing 5 or 10 support images, and one query image. Support images are randomly selected from the training dataset, and the target categories include both positive samples. The support encoder obtains all the category codes.

Our model infers only a fixed number of object queries N in a single pass through the decoder. If the query image x_{query}, and $y_i = (c_i, b_i)_{i=1}^{N}$ is the ground truth of images acquired from the query set, then y_i indicates an object $y_i = (a_i, b_i)$, with a_i being the target category label, and b_i, the bounding box. $y_i = (\varnothing, \varnothing)$ indicates no object. For the support image S_{supp} and its annotation, the detection targets are $y^i = \psi(y_i, c_{supp})_{i=1}^{N}$, where $\psi(y_i, c_{supp})$ acts to filter irrelevant object annotations.

We adopt a pair-wise, or set matching, matching set loss function:

$\mathcal{L}_{match}(y_i^{'}, \hat{y}_{(}\sigma_i))$

to search for a bipartite matching (see equation 4.1) separating $\hat{y}$ and $y^{'}$ against the minimum cost $\hat{\sigma}$,

$$\hat{\sigma} = argmin \sum_{i=1}^{N} \mathcal{L}match(y_i^{'}, \hat{y}\sigma_i) \tag{4.1}$$

with σ the order, or group combination of N image objects, and $\hat{\sigma}$ the best assignment linking outputs and the predictions. We also adopt a cross-entropy loss to categorise the codes produced by support encoder to distinguish between category feature embeddings that belong to different categories.

For the fine-tuning stage, only the maximum epochs set fro training and the learning rate decay epochs were differentiated from the base training. These numbers are empirically set solely based on the training loss trajectory. The setup is presented in Table 4.1.

Table 4.1: Number of epochs and learning rate decay epochs.

	Pascal VOC			**MS COCO**	
	2 shots	5 shots	10 shots	5 shots	10 shots
Total epochs	600	500	500	500	500
Decay epochs	500	380	420	380	420

Inference. With a specified query image, our model produces 100 predictions for each support/query class at inference. We select the best scores predicted throughout all the classes as the eventual predictions.

4.4 Implementation Details

ResNet-101 is selected as a model for the extraction of features for both the support encoder and the query encoders. The model architecture and hyperparameters of the encoder and the decoder remain the same as DETR [20]. After the transformer decoder, we include a feed-forward network (see Figure 4.4). This consists of a 3-layer multilayer

perceptrons (MLP) for box prediction and one layer MLP for object confidence prediction. The model was trained using ADAM optimiser with initial learning rate of 2×10^{-5} and a decay of 1×10^{-5}, batch size of 32. We experimented with 2, and both 5 and 10 images for each query image. We report performance results on both 5 and 10 images. Various data augmentations were adopted during training, including rotation, shifting, and scaling. We follow the existing settings in previous methods [248, 252, 263] methods to evaluate our model on the Pascal-5^i and COCO-20^i datasets. Following [278, 293], the model was trained for 100 epochs for both Pascal-5^i and COCO-20^i. In few-shot settings, the same settings are applied to train the model until convergence. For a fair comparison, we use the same 3 different base/novel splits and a fixed list of novel samples as provided by [101].

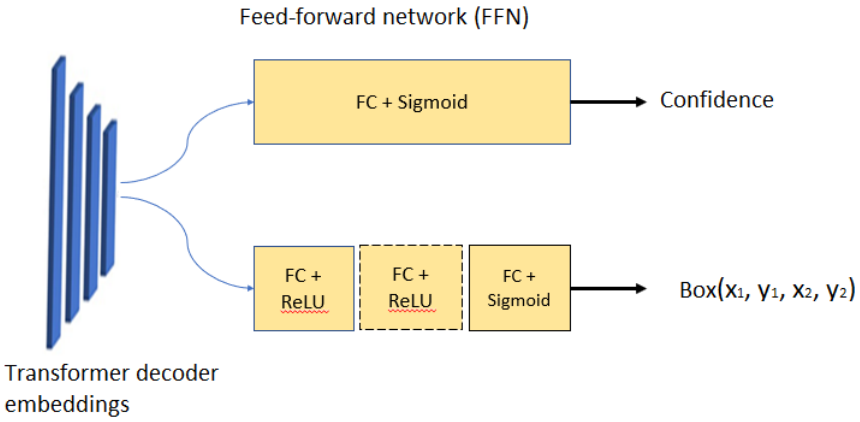


Figure 4.4: The shared decoder feed-forward network (FFN) to produce final predictions. FFN is shared for all the embeddings generated from the transformer decoder.

4.4.1 Model evaluation

We use Pascal-5^i and COCO-20^i datasets for training and inference purposes. For Pascal-5^i, mean average precision (mAP) [173] at IoU threshold 0.5, 0.75 and 0.95 used for Pascal VOC is also used as the evaluation metric. MS COCO's standard metrics, that

include $mAP^{IoU=0.5}$, $mAP^{IoU=.75}$, mAP^{medium} and mAP^{large} are used for evaluation for COCO-20^i, in addition to average precision (AP) and average recall (AR) that measure the percentage of detected objects among all ground truth objects. The AP and AR (see Figures 4.6 and 4.7) are reported for each of the images used, and it is impossible to report all of them in this chapter. The MS COCO metrics also evaluate the performance for objects of different sizes (small, medium, and large). Following [178, 248, 258, 278, 293], who realized that the model performance often relies heavily on the quality of the training samples for novel categories , and that the results come with a large variance, our results are averaged over multiple repeated runs with different randomly sampled support datasets on both Pascal-5^i and COCO-20^i.

```
Accumulating evaluation results...
DONE (t=0.77s).
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.602
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.778
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.650
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.344
```

Figure 4.5: Few-shot model object detection performance on PASCAL-5^i. The results shown here indicate the performance of the model when evaluated with the groundtruth.

The Pascal VOC dataset has object annotations of 20 categories of images. Following [258, 263], we use Pascal-5^i for training and to perform evaluations. We use "3 novel / base category splits, i.e., ("bird", "bus", "cow", "motorbike", "sofa" / others); ("aeroplane", "bottle","cow","horse","sofa" / others) and ("boat", "cat", "motorbike","sheep", "sofa" / others)" [263, p. 1] as in [263]. The number of shots is set to 5 and 10. Mean average precision (mAP) at IoU threshold 0.5 is used as the minimum evaluation metric. Results are averaged over 10 randomly sampled support datasets. MS COCO [263] contains 80 categories including those 20 categories in Pascal VOC. We adopt the 20 shared categories as novel categories, and adopt the remaining 60 categories used in COCO-20^i dataset as base categories for training, and performing evaluations. Standard evaluation metrics for MS COCO, i.e. the COCO Challenge are adopted. Results are averaged over 5 randomly sampled support datasets.

Table 4.2: Summary of object categories used in each fold for the COCO-20^i benchmark datasets.

Dataset	Test categories
COCO-20^0	Person, Airplane, Boat, Park Meter, Dog, Elephant, Backpack, Suitcase, Sports Ball, Skateboard, Wine Glass, Spoon, Sandwich, Hot Dog, Chair, Dining Table, Mouse, Microwave, Fridge, Scissors
COCO-20^1	Bicycle, Bus, Traffic Light, Bench, Horse, Bear, Umbrella, Frisbee, Kite, Surfboard, Cup, Bowl, Orange, Pizza, Couch, Toilet, Remote, Oven, Book, Teddy
COCO-20^2	Car, Train, Fire Hydrant, Bird, Sheep, Zebra, Handbag, Skis, Baseball Bat, Tennis Racket, Fork, Banana, Broccoli, Donut, Potted Plant, TV, Keyboard, Toaster, Clock, Hairdrier
COCO-20^3	Motorcycle, Truck, Stop Sign, Cat, Cow, Giraffe, Tie, Snowboard, Baseball Glove, Bottle, Knife, Apple, Carrot, Cake, Bed, Laptop, Cellphone, Sink, Vase, Toothbrush

```
+ DetectionBoxes_Precision/mAP@.75IOU: 0.649820
+ DetectionBoxes_Precision/mAP@.75IOU: 0.649820
0.343567
+ DetectionBoxes_Precision/mAP (small): 0.343567
+ DetectionBoxes_Precision/mAP (small): 0.343567
 0.602043
+ DetectionBoxes_Precision/mAP (medium): 0.602043
+ DetectionBoxes_Precision/mAP (medium): 0.602043
0.718278
+ DetectionBoxes_Precision/mAP (large): 0.718278
+ DetectionBoxes_Precision/mAP (large): 0.718278

+ DetectionBoxes_Recall/AR@1: 0.476125
+ DetectionBoxes_Recall/AR@1: 0.476125

+ DetectionBoxes_Recall/AR@10: 0.680216
+ DetectionBoxes_Recall/AR@10: 0.680216
+ DetectionBoxes_Recall/AR@100: 0.711243
+ DetectionBoxes_Recall/AR@100: 0.711243
0.424342
+ DetectionBoxes_Recall/AR@100 (small): 0.424342
+ DetectionBoxes_Recall/AR@100 (small): 0.424342
 0.698797
+ DetectionBoxes_Recall/AR@100 (medium): 0.698797
+ DetectionBoxes_Recall/AR@100 (medium): 0.698797
0.789919
```

Figure 4.6: Few-shot model object detection runs on COCO-20^i.

Table 4.3: Few-shot model detection evaluation on Pascal-5^i. We report the standard mAP with IoU threshold 0.5 (mAP50) under 3 different sets of 2 category splits with 5 shots and 10 shots. The results are averaged over multiple repeated runs with different randomly sampled support datasets. Bold indicates the highest model for the 5 shot and 10 shot for each category split.

		Category Split 1		**Category Split 2**		**Category Split 3**	
Method	**Multi-scale**	**5**	**10**	**5**	**10**	**5**	**10**
LSTD [24]	Yes	29.1	38.5	15.7	31.0	27.3	36.3
RepMet [103]	Yes	38.6	41.3	28.3	35.8	34.4	37.2
TFA [248]	Yes	47.9	52.8	34.1	39.5	40.8	45.6
MPSR [247]	Yes	49.4	56.7	36.7	43.3	44.6	50.0
MetaYOLO [251]	No	33.9	47.2	30.1	40.5	42.8	45.9
MetaDet [101]	No	36.8	49.6	31.7	43.0	43.9	44.1
Meta R-CNN [263]	No	45.7	51.5	34.8	45.4	41.2	48.1
Meta-DETR [278]	Yes	**52.2**	**57.8**	44.0	**52.6**	**50.2**	**54.9**
Ours	Yes	51.6	54.7	**44.6**	52.1	50.1	53.8

```
Accumulating evaluation results...
  Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=10  ] = 0.445
  Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=10  ] = 0.631
  Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=10  ] = 0.487
  Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=10  ] = 0.096
  Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=10  ] = 0.442
  Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=10  ] = 0.597
  Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=10  ] = 0.369
  Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=10  ] = 0.515
  Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=10  ] = 0.539
  Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=10  ] = 0.142
  Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=10  ] = 0.531
  Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=10  ] = 0.692
```

Figure 4.7: Few-shot model object detection performance on COCO-20^i. The results shown here indicate the performance of the model when evaluated with the ground truth.

We report the mAP50 of the novel classes on Pascal-5^i with 3 splits in Table 4.3, and compare with the state of the art. In the context of few-shot object detection, mAP is averaged over all novel categories. The table shows the performance for novel categories of Pascal-5^i. Our method performs comparatively with existing methods for most cases. The object detection performance increases with the number of shots, largely attributed to the large search space with image-level predictions. The experimental results demonstrate comparable generalisation ability of our method. Results with base classes not included in this work indicate a far higher performance than that with novel classes. These have been excluded since they are obtained more like using conventional detectors with fine-tuning.

Table 4.4: Few-shot detection performance on COCO-20^i set for novel categories. Unlike Pascal-5^i, we report the mAP averaged over all support image object categories and 10 IoU thresholds (AP@[.5:.05.95]) under 3 different sets of 2 category splits with 5 shots and 10 shots. Results are averaged over multiple repeated runs.

			Average Precision			**Average Recall**		
Shot	**Method**	**Multi scale**	$AP_{0.5}$	$AP_{0.75}$	AP_L	AR_{100}	AR_M	AR_L
5	LSTD [24]	Yes	8.1	2.1	6.5	10.4	5.6	19.6
	TFA [248]	Yes	17.1	8.8	-	-	-	-
	MPSR [247]	Yes	17.9	9.7	16.1	21.2	19.6	34.3
	MetaYOLO [251]	No	12.3	4.6	10.5	14.4	8.4	28.2
	MetaDet [101]	No	14.6	6.1	12.2	15.5	9.7	30.1
	Meta RCNN [263]	No	19.1	6.6	14.0	17.9	5.6	27.2
	Meta-DETR [278]	Yes	28.3	**18.9**	**28.7**	**33.7**	**30.1**	**56.0**
	Ours	Yes	**28.5**	**18.9**	26.8	30.9	29.6	53.8

We also report the performance of our model (see Table 4.4) using some selected metrics of the more challenging COCO Challenge such as $mAP^{IoU=0.5}$ (equivalent to PASCAL VOC metric), $mAP^{IoU=.5}$, mAP^{medium} and mAP^{large} of the COCO novel classes on COCO-20^i with 3 splits in Table 4.3, and compare with the state of the art. In all different base/novel splits, our model achieves a competitive performance for the Average Precision (AP) that directly measures the performance of a detector. As can be seen on the reported 5-shot and 10-shot, our performance is competitive compared to previous state-of-the-art methods, and our results are more inclined to Meta-DETR [278] than the other methods that depend on region-wise object predictions. This demonstrates the importance of DETR's [20] that exploits the effects of global contexts via localisation and classification, largely attributed to the unified image-level meta-learning in our method. It should be noted that the standard COCO Challenge also includes results for $mAP^{IoU=.5:.05:.95}$ averaged over 10 IoU thresholds, mAP^{small}, and other many of their variations that have been excluded to simplify the explanation of the results obtained. The Average Recall (AR) is also an important metric, with higher AR indicating less missed detection. As shown in Table 4.4, our method performs comparatively with the state-of-the-art, and outperforms methods based on region-wise prediction such as Faster R-CNN that rely on region proposals. In contrast, our method eliminates region-wise prediction, and meta-learns object localisation at image level to achieve comparable results with the state-of-the-art.

4.5 Qualitative Results

We provide selected qualitative visualizations of our method's few-shot detection of novel categories results in Figures 4.8 to Figure 4.12 as the major focus is to detect objects of novel categories. In addition, we only show results with confidence scores higher than 0.5. It can be observed that the proposed method is capable of detecting novel objects

even with scarce training samples. In addition, our method performs exceptionally well on large and small objects.

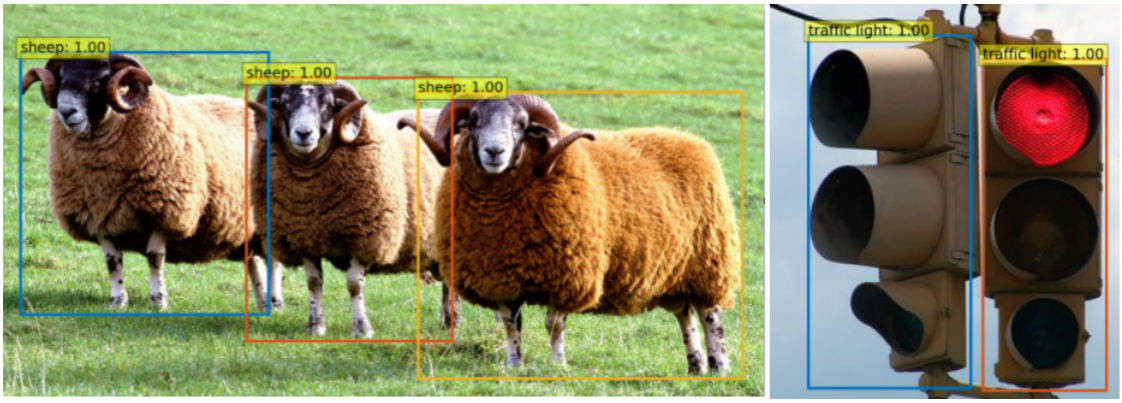


Figure 4.8: Selected qualitative results 1

Figure 4.9: Selected qualitative results 3

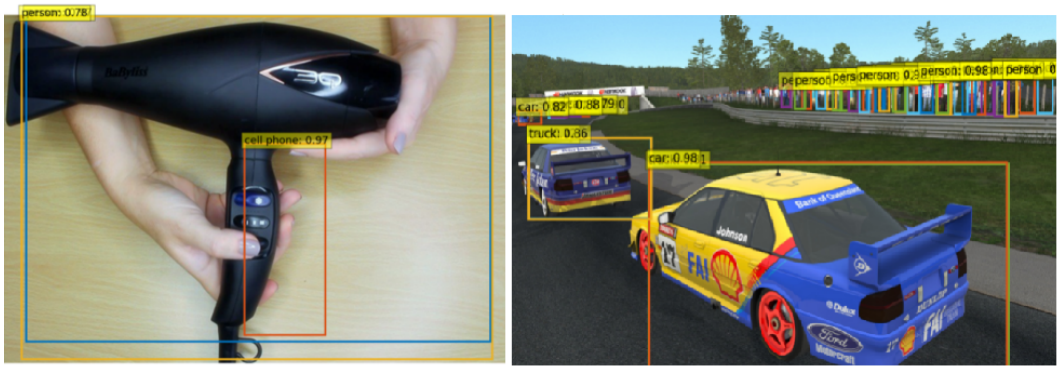


Figure 4.10: Selected qualitative results 4

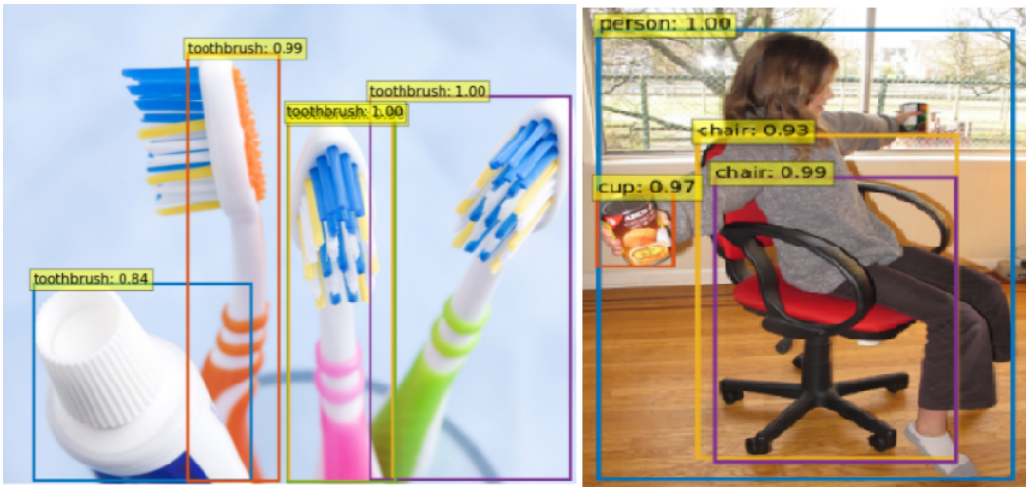


Figure 4.11: Selected qualitative results 5

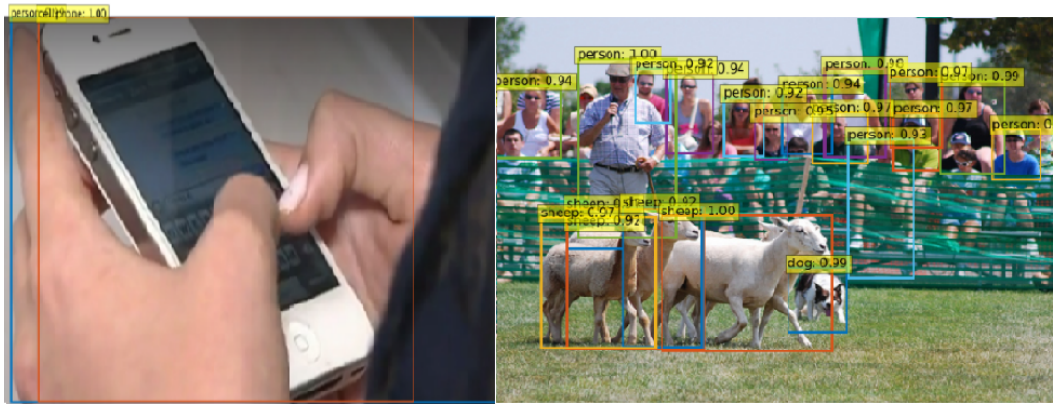


Figure 4.12: Selected qualitative results 6

4.6 Conclusions

This chapter presents a novel few-shot object detection method that meta-learns object localization and classification at the image level. This is achieved by encoding support images and query images into category-specific feature embeddings, and applying a decoder that generates predictions for specific image level predictions of image regions using the transformer. Our approach does not incorporate geometric priors such as non-maximum suppression and anchors, and is fully differentiable. It leverages on the relationship between localization and classification, thereby overcoming the common weaknesses rooted in the region-wise prediction methods. An adopted SAM for aligning high-level and low-level feature embedding semantics is used to improve the generalisation of meta-learned representations. Results from experiments over Pascal-5^i and COCO-20^i object detection benchmarks shows that our method compares favourably with the state-of-the-art in few-shot model for object detection.

In the following chapter, we propose a novel few-shot end-to-end model for "panoptic segmentation" [112, p. 1] that aims to predict and represent foreground objects and background regions using self-attention. The model infers object masks and classes without surrogate tasks and hand-designed components such as bounding box detection and

"non-maximum suppression (NMS)" [20, p. 1] using a dual-path transformer that enables CNNs to read/write a global memory at any layer, and a training objective that optimises a panoptic quality style loss function through "bipartite matching" [20, p. 1] between predicted masks and ground truth masks. Experiments on the Mapillary Vistas dataset demonstrate the effectiveness of the proposed method.

Chapter 5

End-to-End Few-Shot Scene Understanding with Vision Transformer

The previous chapter introduce a novel encoder-decoder approach for few-shot object detection that meta-learns object localisation and classification in a end-to-end manner. Input images from the support and query sets are encoded into feature embeddings that then feed into a category-agnostic decoder that compares the feature embeddings, and generates object predictions for the specific object categories. In this chapter, we build on the previous few-shot object detection chapter, and present a method for scene understanding using a novel few-shot end-to-end panoptic segmentation model that aims to predict and represent objects and background regions in a fully-convolutional backbone that first extracts multiple features in a shared decoder-encoder transformer network following few-shot learning conventions. Our approach uses an object detector from support examples capable of separating target objects from the background thereby resolving class overlaps for non-overlapping segmentation using masks.

5.1 Introduction

"Panoptic segmentation" [112, p. 1] aims to assign individual pixels with a semantic label and unique identity, whereby countable objects, i.e, 'things' and uncountable instances, i.e, 'stuff', are represented and resolved in a unified workflow. The unified representation is made difficult due to their conflicting properties requested by 'things' and 'stuff'. Specifically, countable 'things' generally depend on instance-aware features, which vary with objects, whereas uncountable 'stuff' would generally count on semantically consistent image pixels, which ensures consistent predictions for pixels with the same semantic meaning. The key to solving this few-shot segmentation problem lies in effectively utilizing object information from support examples to separate target objects from the background in a query image. In the end-to-end process, all of the parameters are trained jointly rather than step-by-step. Furthermore, the method uses previously gained input in order to execute its input.

In this chapter, our approach uses an object detector from support examples capable of separating target objects from the background thereby resolving class overlaps for non-overlapping segmentation. We encode object instances or 'stuff' category into a "specific kernel weight with the proposed kernel generator" [130, p. 1] and produce the prediction by convolving the high-resolution feature directly. With this approach, instance-aware and semantically consistent properties for 'things' and 'stuff' can be respectively satisfied in a simple "generate-kernel-then segment workflow" [133, p. 1]. Without extra boxes for instance separation, the proposed approach compares favourably with previous box-based approaches on the Mapillary Vistas [161] dataset with single scale input.

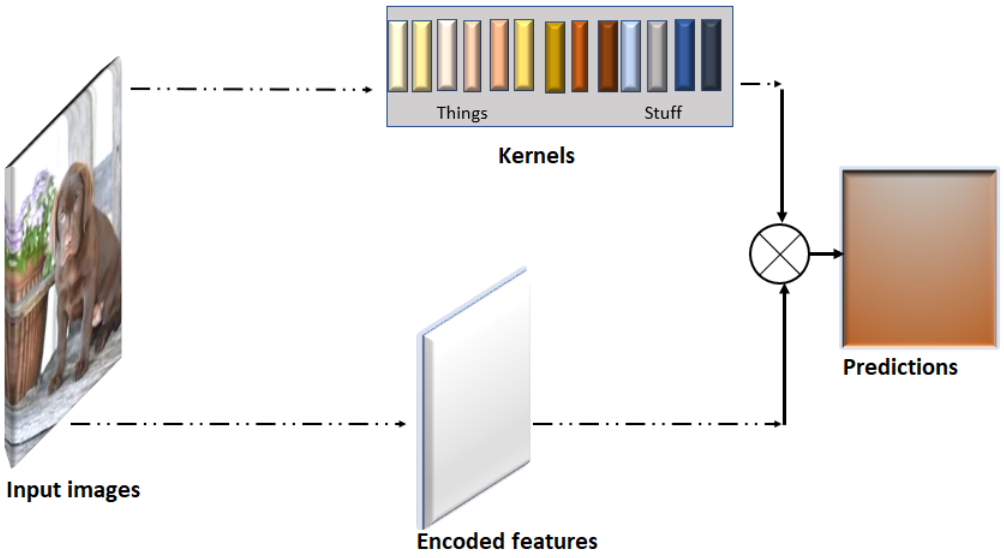


Figure 5.1: We propose end-to-end few-shot panoptic segmentation based on an embeddings generator and a Transformer that represents 'things' and 'stuff' in a unified manner.

The traditional approaches [24, 132, 150, 200] formulate this few-shot segmentation task as a feature matching problem consisting of a support branch and a query branch [123] that apply a CNN to "extract feature maps from their corresponding input images before applying the masked average pooling (MAP)" [283, p. 1] to the support feature map to generate an "object-level representation by pooling the local features over the foreground area specified by the support mask" [283, p. 2]. Finally, this object representation is used to locate target objects in the query image, typically achieved by "pixel-wise similarity comparison between query local features and the object instance representation using" [277, p. 1], for instance, metric representations, and generally used for either instance segmentation, or semantic segmentation.

A drawback is that the object representation produced by the MAP operation might not be able to represent the object well. Local features for different parts will obviously appear differently. Simply pooling over the foreground features may result in a noisy and non-discriminating representation, which further increases the difficulties to locate

target objects in the query image [270]. To solve this problem, we propose an object detector module which learns to produce better-quality object representation that can be integrated into the network and trained end-to-end in few-shot settings. Our qualitative and quantitative results show that object representations generated using our approach are more distinguishable and less noisy.

We incorporate this object detector module into an "encoder-decoder transformer" [20, p. 1] network with several additional modifications to create a powerful and efficient framework for "end-to-end few-shot panoptic segmentation" [94, p. 1]. In particular, our framework first produces object embeddings for support and query images, encodes each instance into a specific kernel and generates the prediction by convolutions directly. Therefore, the "kernel generator and the feature encoder" [133, p. 1] are respectively designed for kernel weights generation and for shared feature encoding. Specifically, in the kernel generator, we draw inspiration from point-based object detectors [292] and utilize the position head to simultaneously locate and classify background 'stuff' by object centres and 'stuff' regions respectively. Then, we select kernel weights with the same positions from the kernel head to represent corresponding instances. A kernel fusion is further proposed for instance awareness and semantic-consistency, which merges "kernel weights that are predicted to have the same identity or semantic category" [133, p. 1]. A feature encoder preserves the high-resolution feature with details. Each prediction of 'things' and 'stuff' can be produced by convolving with generated kernels directly. Thus, both 'things' and 'stuff' can be predicted together with same resolution. In this way, instance-aware and semantically consistent properties for 'things' and 'stuff' can be respectively satisfied in a unified workflow. The key idea is to represent and predict 'things' and 'stuff' uniformly with generated kernels in a fully convolutional pipeline. We evaluate our approach on "Mapillary Vistas" [161, p. 1] benchmarks under both five-shot and ten-shot settings. Experimental results show that our model performs comparatively with the state-of-the-art.

The main contributions of this chapter are:

- We propose a novel few-shot end-to-end model for panoptic segmentation that infers object masks and classes without surrogate tasks and hand-designed components such as box detection, anchors, "thing-stuff merging" [128, p. 1] and "non-maximum suppressions (NMS)" [20, p. 1] using a "dual-path transformer" [243, p. 1] that enables CNNs to read/write a global memory at any layer, and a training objective that optimises a "panoptic quality" [112, p. 1] style loss function through "bipartite matching" [20, p. 1] between predicted masks and annotated masks that depict the ground truth.
- We show that "self-attention mechanisms" [289, p. 1] can be used for few-shot image processing in place of CNNs.
- Experiments on the Mapillary Vistas dataset demonstrate the effectiveness of the proposed method for few-shot panoptic segmentation.

The remainder of this chapter is organized as follows. In Section 5.2, we discuss the related work on the various state-of-the-art few-shot segmentation approaches that were developed recently, including in semantic segmentation, instance segmentation and panoptic segmentation. The proposed few-shot panoptic segmentation method is described in Section 5.3, and the implementation details in Section 5.4. We illustrate the experimental results in Section 5.5. Finally, Section 5.6 concludes the chapter with a summary and an outlook.

5.2 Related Work

Few-shot panoptic segmentation has not been widely studied. Some notable work is available that tackle the problem of panoptic segmentation in situations with vast amounts of

data. Per-pixel classification has been the dominant semantic segmentation since the seminal work of Fully Convolutional Networks (FCN) [150]. For instance, the Atrous Spatial Pyramid Pooling (ASPP) [25, 270] uses atrous convolutions with different atrous rates. Also related to our work in this chapter is Panoptic FCN [133], that encodes 'things' and 'stuff' into a specific "kernel weight using a kernel generator" [133, p. 1], and produces the prediction by convolving feature embeddings directly. They represent and predict foreground instances and background 'stuff' in a unified fully convolutional pipeline without extra boxes for localization or instance separation, with instance-aware and semantically consistent properties for 'things' and 'stuff' that can be respectively resolved in a simple "generate-kernel-then-segment" [133, p. 1] workflow. DANet [65] uses different variants of non-local blocks for instance and semantic segmentation. Recently, DETR [20], MaX-DeepLab [243] and Segmenter [219] replace traditional convolutional backbones with the "Vision Transformer (ViT)" [51, p. 1] that uses self-attention [289], and that capture long-range context starting from the very first layer.

Mask classification has generally been used for instance segmentation that require a dynamic number of predictions. Mask R-CNN [84] uses a global classifier to classify mask proposals for instance segmentation. DETR [20] incorporates a Transformer design to handle 'thing' and 'stuff' segmentation simultaneously for panoptic segmentation. However, these mask classification methods require predictions of bounding boxes, which may limit their usage in semantic segmentation. The recently proposed Max-DeepLab [243] requires multiple auxiliary losses, and removes the dependence on box predictions for panoptic segmentation with conditional convolutions. MaskTransformer [219] proposes a single mask classification model which predicts a set of binary masks, each associated with a single global class label prediction to solve both "semantic segmentation and instance segmentation" [41, p. 1] tasks.

Our pipeline is similar to MaX-DeepLab [243] and DETR [20] that employ a Transformer decoder to compute a set of pairs, each consisting of a mask embedding layer,

and object instance predictions. They also first predict an attention map as raw prediction and use a deep decoder to generate the final prediction and mask results. However, our few-shot learning approach is essentially different from these methods in the way of producing this "similarity attention map" [270, p. 1], and the number of images used during learning. The other main difference is that their methods do not produce "object-level representations" [270, p. 1] from local feature pairs between support and query images. They do not use pair-wise similarity matrix to locate the target object instances in the query image. However, we argue that local features matching are less effective in the few-shot segmentation setting, in which the query and support images are not from the same image and typically look very different. In contrast, our approach focuses on first identification of object instance predictions, and therefore generating better "object-level representations" [270, p. 1] through masks, then using these high-quality object representations to find the target object instances in the query image.

5.3 Proposed Method

5.3.1 Problem definition

Panoptic segmentation segments the image $I \in \mathbb{R}^{H \times W \times 3}$, H = image height, and W = width, into a cluster of categorised masks for the whole:

$\{y_i\}_{i=1}^{K} = \{(m_i, c_i)\}_{i=1}^{K}$,

where K represents the ground truth masks $m_i \in \{0,1\}^{H \times W}$, and do not coincide or encroach into each other, and c_i denotes the prior terrestrial observations that have been made into ground truth class labels of mask m_i.

In few-shot settings, given C_{base}, a dataset of base image categories, and C_{novel}, a set of new categories, few-shot panoptic segmentation aims at detecting objects masks 'stuff' and 'things' of C_{base} by learning from a base dataset D_{base} with abundant annotated instances of C_{base} and a novel dataset D_{novel} with very few instances of C_{novel}. In the

task of K-shot panoptic segmentation, there are K annotated examples from each novel category in D_{novel}.

Inspired by MaX-DeepLab [243] and Meta-DETR [20] on panoptic segmentation, our proposed architecture for few-shot panoptic segmentation includes a dual-path transformer, a stack of decoders, and prediction heads for the masks and the classes.

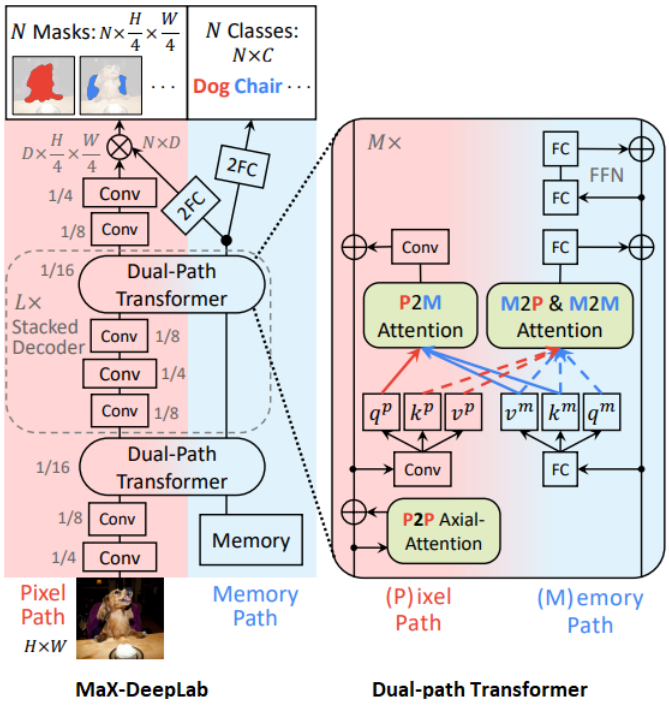


Figure 5.2: Overview of MaX-DeepLab architecture. Image source [243]

Just like in [243, 244], we fuse the transformer and a two-dimensional pixel-based artificial neural network in a dual-path fashion with communication in both directions between the two paths. A two-dimensional CNN combined with a one-dimensional global memory of size N, representing the total number of predictions is augmented to the network model. We also add-on a transformer block for a pre-trained CNN block. The transformer is designed to enable four types of communication between the two-dimensional CNN and the memory path, where each time the query of one is applied to the keys and

values of the other to update either the pixel or memory features conditioned on the other:

- a M2P ("memory to pixel" [243, p. 1]) attention,
- a M2M ("memory-to-memory" [243, p. 1]) self-attention,
- a P2M ("pixel-to-memory" [243, p. 1] feedback attention) that makes it possible for pixels to read from the memory,
- a P2P ("pixel-to-pixel" [243, p. 1] self-attention, will be executed as axial-attention blocks [242, 244], selected instead of the global 2D attention for effectiveness in dealing with the high resolution features.

This transformer design together with a memory path beside the main CNN path is commonly known as dual-path transformer [243]. Consequently, the P2M attention allows the pixel-path CNN to refine its feature given the memory-path features for encoding mask learning and training.

We follow the method used by [243]. We have a two-dimensional input feature $x^p \in \mathbb{R}^{H \times W \times 3}$, height H, width W, and 3 channels; and a one-dimensional global memory feature $x^m \in \mathbb{R}^{N \times 3}$, N the size of the prediction set. We compute pixel-path queries q^p, keys k^p, and values v^p by "learnable linear projections" [51, p. 1] of the pixel-path feature map x^p at each pixel. Similarly, q^m, k^m and v^m are computed from x^m with another set of projection matrices [243]. The query, key and value channels are d_p, and d_v for both paths in the support set, and in the query set. The output of the feedback attention P2M, $y_a^p \in \mathbb{R}^{d_{out}}$, at position a [243], is computed as:

$y_a^p = \sum_{i=1}^{n} softmax_(n)(q_a^p.k_n^m)v_n^m$,

where $softmax_(n)$ denotes a softmax function applied to the whole memory N. Similarly, the output of memory-to-pixel (M2P) and memory-to-memory attention (M2M) attention $y_b^m \in \mathbb{R}^{d_{out}}$ at memory position b is

$y_b^m = \sum_{i=1}^{HW+N} softmax_(n)(q_b^m.k_n^{pm})v_n^{pm}$, with

$k^{pm} = \begin{bmatrix} k^m \\ k^p \end{bmatrix}$, $v^{pm} = \begin{bmatrix} v^m \\ v^p \end{bmatrix}$,

where a single Softmax is performed over the concatenated dimension of size (HW, N).

We use hourglass-style stacked decoders (see Figure 5.3) [200], stacked L times, traversing output strides 4, 8 and 16 multiple times [25] to aggregate multiple scale features. At each decoding resolution, features are joined by some mathematical operation, e.g. summation after resizing, before applying the transformer blocks before the decoded feature is ready for the next resolution.

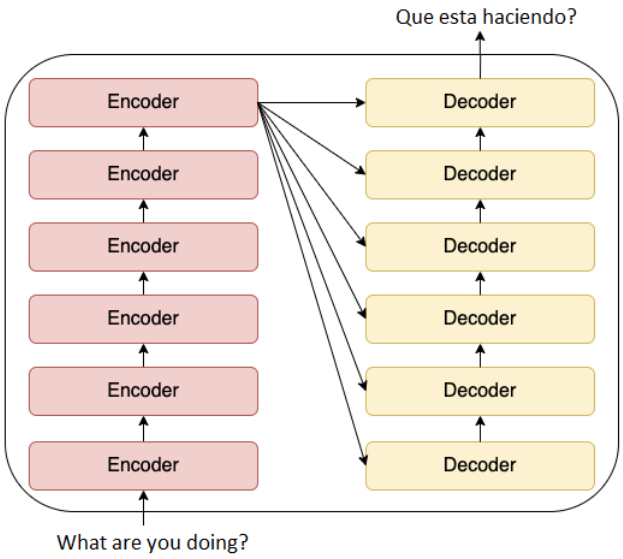


Figure 5.3: Stacked encoders and stacked decoders for used with the transformer.

To predict the masks classes $\hat{p}(c) \in \mathbb{R}^{N \times |\mathbb{C}|}$, we use the memory feature of length N with two FC layers and a softmax, with the other FC head predicting mask features $f \in \mathbb{R}^{N \times D}$. We employ two convolutions to produce a normalised feature $g \in \mathbb{R}^{D \times \frac{H}{4} \times \frac{W}{4}}$ from the decoder input at stride 4. Our mask prediction $\hat{m}$ is the multiplication of the transformer feature f and the decoder feature g, i.e.

$\hat{m} = softmax_N(f.g) \in \mathbb{R}^{D \times \frac{H}{4} \times \frac{W}{4}}$.

The combination $(\hat{m}_i, \hat{p}_i(c))_{i=1}^N$ is our mask transformer output to generate panoptic results.

The mask-prediction head is based on the dynamic and compact instance-aware CondIst [231], conditioned on the instances to be predicted, thereby eliminating the need for ROI cropping and feature alignment.

5.3.2 Losses

We first define a Panoptic Quality (PQ) [112]-style similarity metric between a class-labelled annotated ground truth mask on the image and a model-predicted mask. Then, we demonstrate how we match the predicted mask to each known mask with this metric which is also utilised for optimisation of the model. To demonstrate similarity, we use a metric calculated between the annotated terrestrial mask $y_i = m_i, c_i$, and a prediction mask $\hat{y}_j = (\hat{m}_j, \hat{p}_j(c))$, defined as $sim(y_i, \hat{y}_i) = \hat{p}_j(c_i) \times Dice(m_i, \hat{m}_j)$, equivalent to the multiplication of RQ and SQ, where $\hat{p}_j(c_i) \in [0, 1]$ is the probability of predicting the correct class, and $Dice(m_i, \hat{m}_j) \in [0, 1]$ is the Dice coefficient between a predicted mask $\hat{m}_j$ and a ground truth m_i segmentation quality [243]. The mathematics operation AND gate serves optimises for both model training and mask matching. Zero (0) indicates incorrect prediction of the class, and that the ground truth and the predicted masks do not overlap with each other. One (1) is only achieved when the class prediction is correct, and the ground truth and the mask are the same.

The mask similarity metric we used and the mask matching process is built-based on the PQ-style similarity metric by Max-DeepLab [243]. We optimize model parameters θ by maximizing this same similarity metric over matched (or, positive) masks, i.e.

$\max_\theta \sum_{i=1}^{K} sim(y_i, \hat{y}_{\hat{\sigma}(i)}) \Leftrightarrow \max_{\theta, \sigma \in \mathfrak{S}_N} \sum_{i=1}^{K} sim(y_i, \hat{y}_{\hat{\sigma}(i)})$

Substituting the similarity metric gives our PQ-style objective:

$\mathcal{O}_{PQ}^{POS} = \sum_{i=1}^{K} \hat{p}_{\hat{\sigma}i}(C_i) \mathrm{x} Dice(m_i, \hat{m}_{\hat{\sigma}(i)})$,

Where $RQ = \hat{p}_{\hat{\sigma}i}(C_i)$, and $SQ = Dice(m_i, \hat{m}_{\hat{\sigma}(i)})$

We apply the mathematical product rule and redefine $\mathcal{O}_{PQ}^{POS}$ into two loss terms,of gradient, and then changing the probability $\hat{p}$ to a log probability $log\hat{p}$ which aligns with

the cross-entropy loss and scales gradient better for model efficiency.

$\mathcal{L}_{PQ}^{POS} = \sum_{i=1}^{K} \hat{p}_{\hat{\sigma}i}(C_i).[-Dice(m_i, \hat{m}_{\hat{\sigma}(i)})] + \sum_{i=1}^{K} Dice(m_i, \hat{m}_{\hat{\sigma}(i)}).[-log\hat{p}_{\hat{\sigma}i}(C_i)]$,

where, $[-log\hat{p}_{\hat{\sigma}i}(C_i)]$ is the cross-entropy loss,

$Dice(m_i, \hat{m}_{\hat{\sigma}(i)})$ and $\hat{p}_{\hat{\sigma}i}(C_i)$ are the weights, and

$[-Dice(m_i, \hat{m}_{\hat{\sigma}(i)})]$ the Dice loss, and

where the loss weights are constants. Here, the PQ-style loss is equivalent to optimizing a dice loss weighted by the class correctness and optimizing a cross-entropy loss weighted by the mask correctness [243] so that both of the predicted mask and object class are correct simultaneously. For instance, if the mask is missed, the model must ignore its class since it is a false negative, and vice versa.

Apart from the $\mathcal{L}_{PQ}^{POS}$ for positive masks, we define a cross-entropy term $\mathcal{L}_{PQ}^{NEG}$ for negative unmatched masks:

$\mathcal{L}_{PQ}^{NEG} = \sum_{I=K+1}^{N} [-log\hat{p}_{\sigma i}(\varnothing)]$

This term trains the model to predict $\varnothing$ for negative masks. We balance the two terms by α, as a common practice to weight positive and negative samples [141].

$\mathcal{L}_{PQ} = \alpha\mathcal{L}_{PQ}^{POS} + (1-\alpha)\mathcal{L}_{PQ}^{NEG}$, where $\mathcal{L}_{PQ}$ denotes our final PQ-loss style loss.

We also incorporate "auxiliary losses" [94, p. 3] in training. We use a "pixel-wise instance discrimination loss" [243, p. 3] that helps cluster decoder features into instances. We also use a "per-pixel mask-ID cross-entropy loss" [243, p. 3] that classifies each pixel into N masks, and a semantic segmentation loss. Our total loss function thus consists of the PQ-style loss PQ and these three auxiliary losses as used in [243].

5.3.3 Instance discrimination

We also implement a per-pixel instance discrimination loss as used in [27, 35, 108, 150, 257] that discriminate in an unsupervised fashion or with image classes. It is applied to all pixels in the image, to help the "learning of the feature map" [158, p. 8] $g \in \mathbb{R}^{D \times \frac{h}{4} \times \frac{w}{4}}$.

This encourages features from the same pixel to be similar, and those from different instances to be distinct in a contrastive fashion, a requirement for instance segmentation. We compute a normalized feature embedding $i_{i,:} \in \mathbb{R}^D$, given a ground truth mask $m_i \in \{0,1\}^{\frac{H}{4}+\frac{W}{4}}$ for each "annotated mask by averaging the feature vectors inside the mask m_i" [243, p. 6]:

$t_{i,:} = \frac{\Sigma_{n=1} m_{i,h,w} \cdot g_{:,h,w}}{\|\Sigma_{n=1} m_{i,h,w} \cdot g_{:,h,w}\|}$, i = 1, 2, ..., K.

We get K instance embeddings $\{t_i, :\}_{I=1}^{K}$ representing ground truth masks. Then, we let each pixel feature perform an instance discrimination task by identifying which mask embedding it belongs to, as annotated by the ground truth masks. At each pixel, the contrastive loss is:

$\mathcal{L}_{h,w}^{InstDis} = -log \frac{\sum_{i=1}^{K} m_{i,h,w} exp(t_{i,:} . g_{:,h,w}/\mathcal{T})}{\sum_{i=1}^{K} exp(t_{i,:} . g_{:,h,w}/\mathcal{T})}$,

where $\mathcal{T}$ denotes the temperature. $m_{i,h,w}$ is non-zero only when pixel (h, w) belongs to the ground truth mask m_i.

Inspired by [20, 243], we use cross-entropy loss together with a dice loss to learn better segmentation masks to train this per-pixel classification to infer mask-ID maps given by our mask prediction. An auxiliary semantic segmentation loss [243] is used to help capture per pixel semantic features. We apply a semantic head as used in [34] on top of the backbone if no stacked decoder is used, or connect the semantic head to the first decoder output at stride 4.

5.3.4 Mask matching

We solve a "bipartite matching problem" [20, p. 1] between the prediction set $(\hat{y}_i)_{i=1}^{N}$ and the ground truth $\{y_i\}_{i=1}^{K}$ to assign a predicted mask to each ground truth. We search for a permutation of N elements $\sigma \in \mathfrak{S}_N$ that best assigns the predictions to achieve the "maximum total similarity to the ground truth" [243, p. 7]:

$\hat{\sigma} = \text{argmax}_{\sigma \in \mathfrak{S}} \sum_{i=1}^{K} sim(y_i, \hat{y}_{\sigma(i)})$.

Following prior work [20, 243], the optimal assignment is computed efficiently with the Hungarian algorithm [20, 118]. K matched predictions will be optimized to predict the corresponding ground truth masks and classes, while the remaining N–K masks predicts the absence of an object. Only one predicted mask can be matched, with an IoU over 0.5 with each ground truth mask.

5.3.5 Network description

In few-shot learning, the goal of training is to learn the similarities and differences between objects. Our few-shot end-to-end model for panoptic segmentation infers "instance object masks and classes without surrogate tasks such as box detection, anchors, thing-stuff merging" [243, p. 1] and "non-maximum suppressions (NMS)" [20, p. 1] that are designed by hand. MaX-Deeplab [243] utilises "dual-path transformer" [243, p. 1] that enables CNNs to read/write a model's global memory for all the CNN layers, and a training objective that optimises a PQ-style loss function through "bipartite matching" [20, p. 1] between predicted masks and ground truth masks. We employ a shared support branch and a query branch following the philosophy of the "Siamese network" [113, p. 1], and a decoder branch. Following few-shot learning conventions, the support branch handles images in the support set, while the query branch handles images in the query set during training.

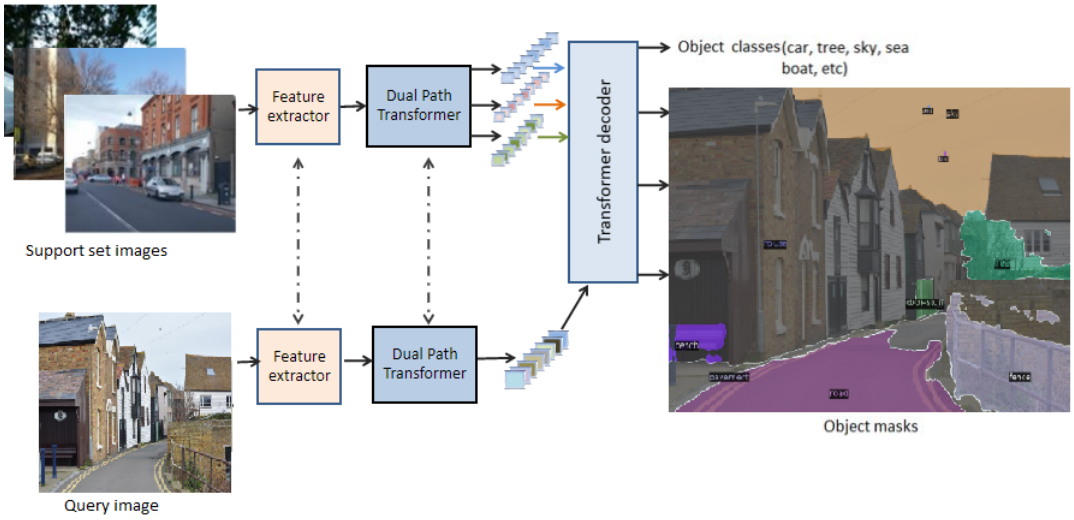


Figure 5.4: Overview of the proposed model for few-shot panoptic segmentation using a ViT.

5.3.6 The Transformer decoder

The decoder follows the standard architecture of the transformer [237], transforming N embeddings of size d using multi-headed self- and encoder-decoder attention mechanisms in parallel. Using the set prediction mechanism proposed in DETR [20], we employ a decoder to compute a set of pairs each consisting of a mask embedding vector to get the binary mask prediction via a dot product with the per-pixel embedding obtained from an underlying fully-convolutional network, and object instance predictions to solve both few-shot semantic segmentation and instance segmentation tasks in a unified manner.

5.3.7 Training and inference

Panoptic segmentation is a challenging task. We train the model first (see Figure 5.4 for the model) on the categories on COCO-20^i few-shot learning dataset. We then train it on the Mapillary Vistas for scene understanding. We only evaluate the model on the Mapillary Vistas dataset. The COCO-20^i dataset include several object instances and 'stuff'

categories. During training, a few images from the support set with instance annotations, and the query image are provided during each episode. The Support branch extract category codes that mostly relate to instance object predictions and mask predictions within the support set images. The Query branch, which aggregates the query features and category codes of the query set into a set of category-specific features, shares all the learning parameters with the Support branch. This Support branch aims at extracting the category codes of objects in the image. Pixels which are not occupied by object categories are labelled as 'stuff'. Both the Query branch and the Support branch first infers masks and classes using the attention mechanism, and encode them into query features and instance object category codes, respectively, using a shared Transformer network. In both branches, we optimize the network in an end-to-end manner using the loss functions described in Section 5.3.2, with other important parameters described in 5.4. The Decoder branch then takes the query features and the instance category codes as input and predicts segmentation masks and the instance object detection predictions results over the corresponding support image masks and instance object categories. The instance target categories to detect are dynamically conditioned on the provided support images. We do not adopt the atrous spatial pyramid pooling module (ASPP) [25], since our attention block could also efficiently encode the multi-scale or global information. In this way, our method is designed to extract category-agnostic meta-level knowledge that can easily adapt to novel image categories.

5.4 Implementation Details

Following Axial-DeepLab [244], MaX-DeepLab [243] and DETR [20] settings, we use a learning rate of 0.1, and a weight decay of 10^{-4}, and a drop-path rate of 0.2. The images are resized to 640×640 for inference and calculations. We adopt a batch size of 64 and each query image is associated with 10 support images to form an episode.

Conventional data augmentation as used in [20, 294] is adopted during training. We set masks with class confidence of 0.70 to void and filter pixels with mask-ID confidence below 0.4. We limit 'stuff' masks to a maximum of 4096 pixels, and 'thing' masks to 256. In training, we set our PQ-style loss weight normalized by N to 3.0, with α = 0.75. Instance segmentation uses τ = 0.3, and weight 1.0, mask-ID cross entropy weight to 0.3, and semantic segmentation weight to 1.0. We use an output size N = 128 and D = 128 channels. We fill the initial memory with learnable weights same as DETR.

5.4.1 Panoptic segmentation datasets

Dataset. We use Mapillary Vistas [161] dataset for traffic-related environments which has resolutions ranging from 1024×2048 to 4000×6000. This dataset contains images from all around the world that have been captured at various conditions regarding weather, season and daytime. Images come from different imaging devices, including mobile phones, action cameras, and professional capturing rigs by differently experienced photographers. It has 25,000 images pixel-accurately labelled into 66/124 object categories of which 37/70 classes are instance-specific labels with 37 'thing' classes and 28 'stuff' classes. Annotation is performed in a dense and fine-grained style by using polygons for delineating individual objects.

5.5 Experimental Results

Our experiments were conducted on the Mapillary Vistas dataset for traffic-related environments. We compare our panoptic segmentation results with other box-based and box-free methods in Table 5.1. Our model compares favourably with the state-of-the-art, and can be improved with simple enhancements. It attains a PQ of approximately 43% (see Figure 5.5).

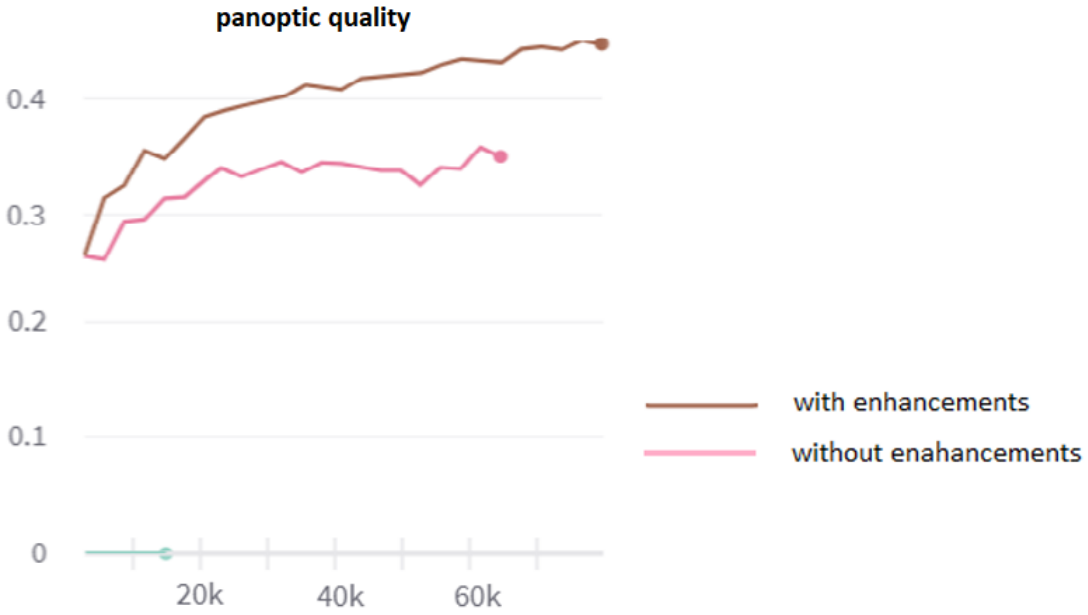


Figure 5.5: Panoptic quality for Mapillary Vistas.

5.5.1 Qualitative Results

We provide qualitative results of our few-shot panoptic segmentation of novel object categories (‘things’, ‘stuff’) in Figures 5.6 to Figure 5.10. These visualisations indicate that our proposed method is capable of detecting novel ‘things’ and ‘stuff’ of novel categories objects even with scarce (n=5) training samples, and even performs exceptionally well on both small and large objects.

Table 5.1: Few-shot panoptic segmentation experiments on the Mapillary Vistas dataset. All our results are achieved with single scale input. Some results courtesy of [131]

Method		**Backbone**	PQ	PQ^{th}	PQ^{st}
UPSNet [261]	box-based	DCN101-FPN	46.6	53.2	36.7
SpiNet [94]	box-based	DCN101-FPN	42.2	49.3	31.4
Panoptic FPN [111]	box-based	Xception-71	40.9	48.3	29.7
SOGNet [265]	box-based	DCN101-FPN	**60.0**	56.7	**62.5**
CIAE [69]	box-based	DCN101-FPN	44.5	49.7	36.8
BANet [30]	box-based	DCN101-FPN	47.3	54.9	35.9
Panoptic FCN [133]	box-based	ResNet101-FPN	45.5	51.4	36.4
DeeperLab [264]	box-free	Xception-71	34.3	37.5	29.6
Axial-Deeplab [244]	box-free	Axial-ResNet-L	43.6	48.9	35.6
Panoptic DeepLab [34]	box-free	Xception-71	39.7	43.9	33.2
DetectoRS [184]	box-free	ResNet101	49.6	**57.8**	37.1
Detectron2 [256]	box-free	ResNet101	43.0	–	–
DETR [20]	box-based	CNN	45.1	50.5	37.0
Ours	box-free	ResNet 101	44.1	45.3	34.6

Figure 5.6: Panoptic segmentation qualitative results 1.

Figure 5.7: Panoptic segmentation qualitative results 2.

Figure 5.8: Panoptic segmentation qualitative results 3.

Figure 5.9: Panoptic segmentation qualitative results 4.

Figure 5.10: Panoptic segmentation qualitative results 5.

5.6 Conclusion

We have presented novel few-shot end-to-end panoptic segmentation method that aims to predict and represent objects and background regions in a fully-convolutional backbone based on Max-DeepLab in this chapter. The model first extracts multiple features in a shared decoder-encoder, and uses an object detector from support examples capable of separating target objects from the background thereby resolving class overlaps for non-overlapping segmentation using masks. In the following chapter, we propose a two-stage training regime in few-shot knowledge distillation that utilises self-supervised contrastive learning to enforce diversification limitations on output space. We use supervised contrastive learning for learning the image feature embeddings that will be used in the second stage to leverage the label information to further fine-tune the model to perform classification using a distillation loss.

Chapter 6

Contrastive Self-supervised learning with Knowledge Distillation for Few-shot Image Classification

The previous chapter presents a few-shot end-to-end panoptic segmentation method that aims to predict and represent objects and background regions in a fully-convolutional backbone using a Transformer encoder-decoder. This chapter contributes to improve the representation capabilities of few-shot learning models using self-supervised learning. We follow a two-stage training regime in few-shot knowledge distillation that utilises self-supervised contrastive learning to enforce multiplicity constraints in the image output space. We use supervised contrastive learning for learning the image feature embeddings that will be used in the second stage to leverage the label information to further fine-tune the model to perform classification using a distillation loss.

6.1 Introduction

Although CNNs have achieved breakthroughs using supervised learning on large-scale datasets such as ImageNet, they often fail to generalise when the dataset is small, resulting in over-fitting. The world has overwhelmingly large number of object classes and varying levels of abundance that makes them impossible to annotate enough examples, and to learn at once, making few-shot learning mandatory in most practical situations. Regularisation approaches such as dropout, batch normalisation [95], and augmentation involving diverse transformation strategies that have previously been proposed have not entirely solved over-fitting problems in situations with limited data. Self-supervised learning has shown potential of learning useful representations from data without external label information. In particular, the contrastive learning methods [38, 85, 93, 229] have demonstrated potential in self-supervised learning methods. They learn better, unbiased, transferable representations for downstream tasks which can effectively prevent the model from over-fitting.

Self-supervised learning focusses on obtaining good representations of data where human interaction is eliminated, and data labelling is automated[168, 169, 170]. For instance, in computer vision, "representations can be learned from predicting transformations" [26, p. 1] , generative modelling [75], and other techniques. Recently, self-supervised representation learning algorithms, e.g. by [81, 85, 86] with contrastive loss have performed well in extracting useful representations. The key idea of contrastive learning is to contrast semantically positive and negative pairs of images, encouraging the representations f of similar pairs $(x, x+)$ to be close, and those of dissimilar pairs $(x, x–)$ to be more orthogonal [38].

Knowledge distillation (see Chapter 2), proposed by [89] trains a smaller network known as the student using the supervision signals from both ground truth labels and either an ensemble of models or well-learned larger model called the teacher by using the predicted logits for the transfer of knowledge. The student model is trained to mimic the

prediction capabilities of the teacher. It is the one which is used during testing, prediction and deployment. In self-distillation [125], the same model is used both for teacher and student as a regulation term to prevent the model from over-fitting.

In this chapter, we build on this insight and explore the capability of contrastive learning to make better the metric representation potential of CNNs for few-shot learning in two stages. During Stage One, we utilise self-supervised contrastive learning for learning image feature embeddings before using the image labels to train a model in a self-supervised manner. Stage Two uses the weights obtained from the Stage One and leverage the label information to fine-tune the model for better classification. Our method basically uses a self-supervised loss to train a CNN to augment the entropy of the feature embedding. This results in lowering the entropy on feature representations by driving self-supervised pairs closer, and simultaneously compel the manifold with student-teacher knowledge distillation [188] in Stage Two. Our work is related to knowledge distillation methods [210, 215, 262, 280, 295] that distils the knowledge of an ensemble of a large teacher model to a smaller student neural network at the classifier.

The main contributions of this chapter are:

- We propose a training regime in few-shot knowledge distillation that utilises self-supervised contrastive learning to enforce multiplicity restrictions during the output space. We use supervised contrastive learning for learning the image feature embeddings that will be used in the second stage to leverage the label information to further fine-tune the model to perform classification using a distillation loss.

- Experimental results show that our model achieves comparable few-shot classification performance compared to existing state-of-the-art methods on both MiniImageNet and CIFAR-FS benchmarks.

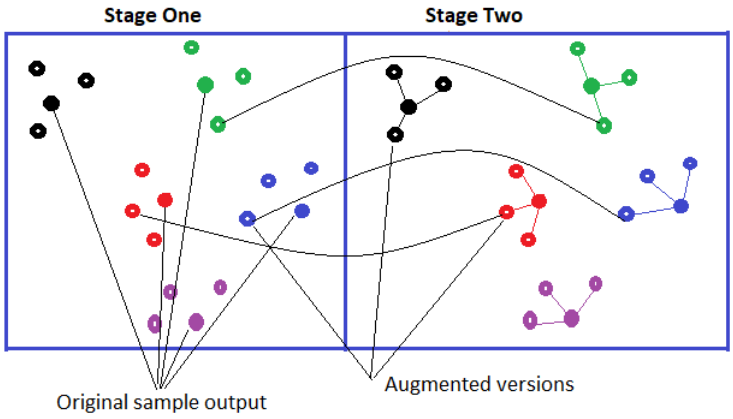


Figure 6.1: Self-supervised learning with knowledge distillation. The true prediction manifold is estimated using self-supervision during Stage One which is equivariant to input transformations by enforcing the model to foretell the number of augmentations utilising the produced logits. In Stage Two, the existing sample outputs are enforced to be identical to those in Stage One, while minimising the distance with the augmentations.

The remainder of this chapter is organized as follows. In Section 6.2, we discuss the related work on the various state-of-the-art in self-supervised few-shot knowledge distillation approaches that were developed recently. The proposed two-stage self-supervised knowledge distillation for few-shot learning is described in Section 6.3. We illustrate the datasets and discuss experimental results in Section 6.5. Finally, Section 6.7 concludes the chapter with a summary and an outlook.

6.2 Related Work

6.2.1 Self-supervised learning for few-shot learning

Self-supervised learning is a paradigm for unsupervised learning that aims to learn robust representations from the data itself without any manual class annotations, or labels. The main challenge here is how to design the annotation-free pretext tasks, including rota-

tions [71, 199], relative patch locations [32, 71], and colourisations [125, 281] that are complex enough to exploit high-level semantic visual representations useful for solving downstream tasks.

Recently, the potential of self-supervised learning for few-shot learning was explored in [71, 188, 199, 220]. Gidaris et al. [71] uses self-supervision as an auxiliary task in few-shot learning for feature extraction to learn richer, more transferable visual representations in an attempt to improve transfer learning abilities of few-shot models. They propose a two-stage learning approach that adds a self-supervised loss to the training loss that a few-shot model minimises in the first stage. A parallel branch with artificially augmented tasks help the model learn generalizable features and adapt to novel classes with few training data. Simultaneous equivariance and invariance by [199] allows the model to learn independent input transformations, as well as the features that encode the structure of geometric transformations of an image that generalise well to novel classes. Su et al. [220] also used rotation and permutation of patches as auxiliary tasks and concluded that self-supervised learning is more effective in low-data regimes, and that performance improvements are mainly realised when images used are within the same domain as the base classes. They, therefore, propose an approach that picks similar-domain unlabelled images. SimCLR ("Simple Framework for Contrastive Learning of Visual Representations" [27, p. 1]) presents a framework that learns representations by maximizing agreement between augmented pairs of data via a contrastive loss in either unsupervised pre-training or episodic training for few-shot learning. They show that effective representations are achieved by compositions of data augmentation operations such as "random cropping followed by resize back to the original size, random colour distortions, and random Gaussian blur" [27, p. 2].

6.2.2 Knowledge distillation

Among the many different forms of distilling knowledge that have been utilised are distilling knowledge from logits [68, 89, 156], distilling knowledge from intermediate layers [126, 268], knowledge distillation with meta-learning [109], and knowledge distillation with self-supervised learning [188].

Notable work in self-supervised "few-shot knowledge distillation" [112, p. 1] includes who introduce a "dual-stage distillation" [210] scheme that grafts each one of the student blocks onto the teacher, and learns the parameters of the grafted block intertwined with those of the other teacher blocks in the first step. In the second step, the trained student blocks are progressively connected and then together grafted onto the teacher network, getting to adapt themselves to each other and eventually replace the teacher network. Bai et al. [8] propose a network-compression layer-wise cross knowledge distillation approach by interweaving hidden layers of teacher and student network thereby reducing accumulated estimation errors and improving predictions. Teacher Assistant Knowledge Distillation (TAKD) [156] introduce "multi-step knowledge distillation" [156, p. 1] which employs an intermediate-sized teacher assistant network to bridge the gap between the student and the teacher. They argue that the performance degrades when the gap between student and teacher is large, that the teacher can only effectively transfer its knowledge to students up to a certain size. Self-supervised knowledge distillation (SKD) [188] learn feature embeddings to create various input-output spaces using a "self-supervised auxiliary loss" [48, p. 1]. They lower the entropy of the feature representations by guiding self-supervised pairs jointly all together while limiting the manifold with student-teacher knowledge distillation.

Existing approaches disregard the importance of intra-class diversity, and instead focus on inter-class representations. Unlike [199], we argue for equivariant representation to learn the natural manifold of the class. Major input transformations are desired in corresponding outputs to ensure diversity. In this work we are interested in knowledge

distillation with self-supervised learning, which involves a bigger teacher network during training and fine-tuning on unlabelled data in Stage One. During Stage Two, the bigger network can be distilled into a smaller network with little or no loss in accuracy. In contrast to the existing self-supervised learning approaches for few-shot learning, our approach employs self-supervision to compel additional constraints in its output space for the classification. We learn an equivariant representation to learn an object class in few-shot learning settings. We expect that, in a few-shot setting, where squeezing out "generalizable features from the available data is very important, the use of self-supervision as an auxiliary task will bring improvements" [19, p. 1].

6.3 Proposed Method

A two-stage approach that employs "self-supervised learning" [147, p. 1] with "knowledge distillation" [18, p. 1] is proposed, 1) self-supervised pre-training Stage One, and 2) Stage Two that uses the original image as anchors to maintain the acquired manifold. The Stage One uses self-supervision to learn a broader classification manifold, and that the acquired representations are equivariant to the augmentations. At this stage, it is important that the augmented image samples, together with the original inputs are reinforced to have similar prediction results to further improve the quality of between-class discrimination. The learned representations can be considered as visual priors before using the label information. Stage Two is used to initialize both the teacher and student model used in the self-distillation process with the pre-trained weight. During this stage, the first self-supervision stage is used as a teacher, and the original, non-augmented images are used as anchors to maintain the learned manifold. Augmented image labels are applied to minimise intra-class distance representations in the embedding output space to learn robust image feature representations that can be used for image discriminations. The weight of the teacher is frozen, and the student is updated using a combination of the classifi-

cation loss and the overhaul-feature-distillation loss from the teacher. Consequently, the student model is regularized by the representation from the teacher when performing the classification task.

The teacher-student knowledge distillation guides the model to develop two inherent attributes. In the first place, the output class manifold is divergent enough to preserve major transformations in the input space. This, therefore strengthen the avoidance of over-fitting, resulting in improving generalization capabilities of the model. Second, the learned embeddings associations in the output space encode natural associations among close classes that should have correlated predictions as opposed to different image classes. Thus, the model portrays the representation space by learning "inter-class relationships and preserving intra-class diversity" [188, p. 2], thereby learning improved representations for few-shot learning tasks.

6.3.1 Network description

We have a CNN $\mathcal{F}$ with feature representations Φ and weights Θ for classification. The input image x is mapped to a vector representation $v \in \mathbb{R}^d$ by a function $\mathfrak{f}_\Phi : \mathrm{x} \rightarrow \mathrm{v}$. Feature vector representations v are then mapped to logits $p \in \mathbb{R}^c$ by the function $\mathfrak{f}_\Theta : \mathrm{v} \rightarrow \mathrm{p}$, and c represents the quantity of resulting classes. Therefore, $\mathcal{F} = \mathfrak{f}_\Phi \circ \mathfrak{f}_\Theta$, i.e. composition of the two functions. We introduce $\mathfrak{f}_\Psi$, so that $\mathfrak{f}_\Psi : \mathrm{p} \rightarrow \mathrm{q}$, which is mapped to logits p to an auxiliary set of logits $q \in \mathbb{R}^s$ for the augmented task of self-supervised. Every input x produce labels $r \in 1, \ldots, s$ used for self-supervision. Thus, the entire model network as employed in [188] is represented as:

$$\mathcal{F}_{\Phi,\Theta,\Psi} = \mathfrak{f}_\Phi \circ \mathfrak{f}_\Theta \circ \mathfrak{f}_\Psi.$$

The network also has $\mathcal{D}_{base}$, a dataset with n image-label pairings $\{x_i, y_i\}_n$, whereby $y_i \in \{1, \ldots, c\}$. Following few-shot learning settings (See Chapter 2), we sample episodes

using (S), i.e. labelled support set, and (Q) the query set. In an n-way, k-shot setting, (S) has k number of examples for every individual n class.

6.3.2 Stage One

We adopt the training methods used in [155, 188, 199]. A mini-batch $\{x, y\}$ is selected randomly from D_{base} with m numbers of image-label pairs during Stage One, so that $\{x, y\} = \{x_i, y_i\}_m$. A transformation function $\mathcal{T}(.)$ [47, 125] is applied to images x to create four suitable augmentations [125, 269, 275], e.g. by colourisation, or/and applying rotations to the image x. All the augmentations are combined into a single tensor $\hat{\mathbf{x}}$ with matching class category labels $\hat{\mathbf{y}} \in \mathbb{R}^{4\mathrm{x}m}$, and one-hot encoded labels $\hat{r} \in \mathbb{R}^{s}_{\{4\mathrm{x}m\}}$ for the type of augmentation, where $s = 4$, a result of the 4 augmentations for self-supervised.

The augmented tensor $\hat{\mathbf{x}}$ is passed via the function $\mathfrak{f}_\Phi$, producing the feature representations $\hat{\mathbf{v}} \in \mathbb{R}^{d\mathrm{x}(4\mathrm{x}m)}$, that then go past $\mathfrak{f}_\Theta$ resulting in matching logits $\hat{\mathbf{p}} \in \mathbb{R}^{c\mathrm{x}(4\mathrm{x}m)}$. The logits finally go past $\mathfrak{f}_\Psi$, to get the augmentation logits $\hat{\mathbf{q}} \in \mathbb{R}^{s\mathrm{x}(4\mathrm{x}m)}$,

$$\mathfrak{f}_\Phi(\hat{\mathbf{x}}) = \hat{\mathbf{v}} \qquad \mathfrak{f}_\Theta(\hat{\mathbf{v}}) = \hat{\mathbf{p}} \qquad \mathfrak{f}_\Psi(\hat{\mathbf{p}}) = \hat{\mathbf{q}}$$

To optimise the model in the Stage One, we employ a categorical cross entropy loss used in [188] $\mathcal{L}_{ce} = -log\left(\frac{e^{p_y}}{\sum_j e^{p_j}}\right)$ between the estimated logits $\hat{\mathbf{p}}$ and $\hat{\mathbf{y}}$, and a self-supervision loss $\mathcal{L}_{ss} = -log\left(\frac{e^{q_r}}{\sum_j e^{q_j}}\right)$, i.e. a a binary cross entropy loss, between the augmented logits $\hat{\mathbf{q}}$ and their labels $\hat{\mathbf{r}}$. We also combine the two losses with a weighting coefficient α to obtain the final loss,

$$\mathcal{L}_{StageOne} = \mathcal{L}_{ce}(p, y) + \mathcal{L}_{ss}(q, r).$$

The whole training process for **Stage One** is stated as the following optimisation prob-

lem:

$$\min \Phi, \Theta, \Psi \; \mathbb{E}_{x,y\sim\mathcal{D}}[\mathcal{L}_{ce}(f_{\Phi,\Theta}(\bar{\mathbf{x}}), \bar{\mathbf{y}}) + \alpha.\mathcal{L}_{ss}(f_{\Phi,\Theta,\Psi}(\bar{\mathbf{x}}), \bar{\mathbf{r}})]$$

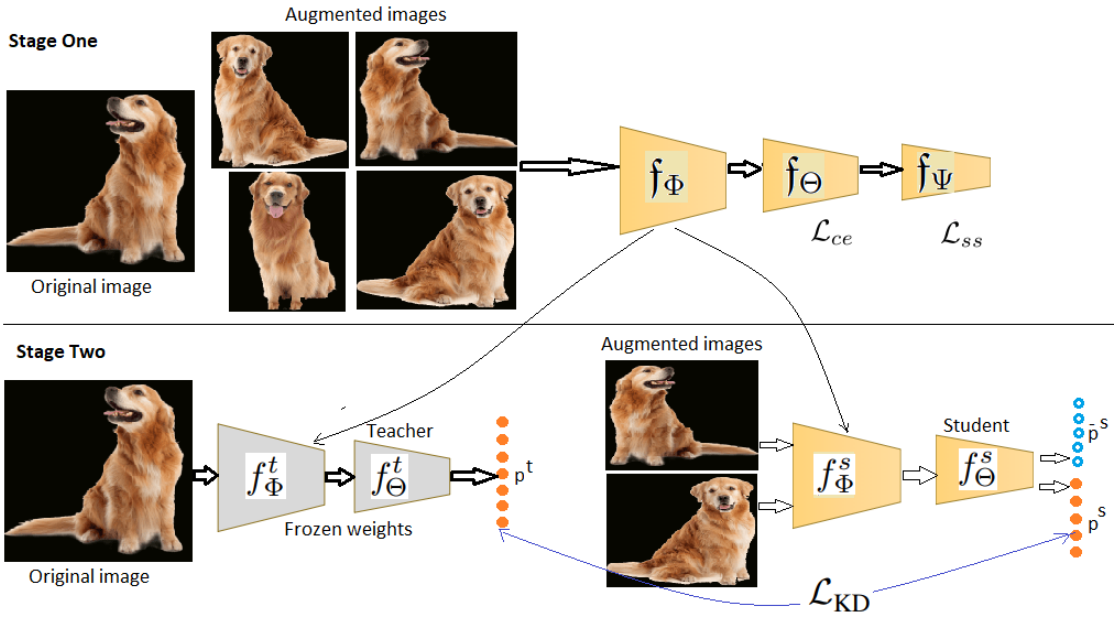


Figure 6.2: Illustration of the our self-supervised knowledge distillation model for training with augmented image versions to predict the class, then using the original version of the images to maximise the discriminative capability of our model.

6.3.3 Stage Two

During Stage Two, we take the teacher F^t model that teaches the student F^s learner model following approaches by [89, 126, 262, 269], the two clones of the model trained during the Stage One. We freeze the weights of the bigger teacher model, and use it for the purposes of inference. Our approach starts by sampling a mini-batch (x, y) from $\mathcal{D}_{base}$ and create a pair $\bar{x} \in \hat{x} \setminus x$ from x, such that $\bar{x}$ is an augmented version of x. x is employed as a focal point to regulate any changes to the classification manifold compelled by a knowledge distillation loss between the two clone networks. We also

employ an auxiliary l_2 loss to guide the image embeddings of x and $\bar{x}$ in conjunction to enhance feature differentiation, and their discriminability. Following [188], we pass x via the teacher network $F^t = f^t_{\Phi,\Theta} \circ f^t_{\Psi}$ and its logits $\mathbf{p}^t$ are acquired. $\mathbf{x}$ and $\bar{\mathbf{x}}$ are then passed via F^s to obtain their logits $\mathbf{p}^s$, and $\bar{\mathbf{p}}^s$ respectively.

$f^t_{\Phi,\Theta}(\mathbf{x}) = \mathbf{p}^t, \qquad f^s_{\Phi,\Theta}(\{\mathbf{x}, \bar{\mathbf{x}}\}) = \{\mathbf{p}^s, \bar{\mathbf{p}}^s\}, \qquad f_{\Phi,\Theta} = f_{\Theta} \circ f_{\Phi}$

We minimise the Kullback–Leibler divergence [119], or relative entropy, to measure the difference between the probability distributions between $\mathbf{p}^t = \{\mathbf{p}^t_i\}$ and $\mathbf{p}^s = \{\mathbf{p}^s_i\}$ for knowledge distillation, and carry out an $\mathcal{L}_2$ loss between $\mathbf{p}^s$ and $\bar{\mathbf{p}}^s$ to achieve better discriminability,

$\mathcal{L}_{KD}(\mathbf{p}^s, \mathbf{p}^t, T) = \mathbf{KL}(\sigma(\mathbf{p}^s/T), \sigma(\mathbf{p}^t/T)), \qquad \mathcal{L}_{l_2} = \| \mathbf{p}^s - \bar{\mathbf{p}}^s \|_2,$

with σ being a softmax function and T is a temperature parameter [89] used to soften the output distribution. The two losses are combined by a coefficient β as follows,

$\mathcal{L}_{StageTwo} = \mathcal{L}_{KD} + \beta.\mathcal{L}_{l_2}$

The overall **Stage Two** training process can be formalised as the following optimization problem:

$\min\{\Phi, \Theta\}\ \mathbb{E}_{x,y\sim\mathcal{D}}[\mathcal{L}_{KD}(f^s_{\Phi,\Theta}(\mathbf{x}), (f^t_{\Phi,\Theta}(\mathbf{x})) + \beta.\mathcal{L}_{l_2}(f^s_{\Phi,\Theta}(\mathbf{x}), (f^s_{\Phi,\Theta}(\bar{\mathbf{x}}))]$

6.4 Evaluation

Following few-shot learning, we evaluate our model for classification only using a support set $D_{support} = \{x_{support}, y_{support}\}$ and a query set D_{query} from the held-out part of the dataset D_{base}, while $D_{query} = x_{query}$. Both are fed to the final trained f^s_{Φ} model to get $v_{support}$ and v_{query}, i.e. the feature embeddings, respectively. To map the labels $D_{support}$ to D_{query}, a logistic regression classifier [230] is used. We normalise the embeddings onto a unit sphere [15]. We randomly sample 400 tasks, and report mean classification accuracy with 95% confidence interval.

6.5 Experiments and Results

We evaluate our method on the classification of images on two benchmark few-shot learning datasets, the MiniImageNet [190, 239] and CIFAR-FS [15, 116]. We use the split proposed in [190], with 64, 16 and 20 classes for training, validation and testing for MiniImageNet. CIFAR-FS is randomly sampled from CIFAR-100 [116] by using the same criteria with which MiniImageNet has been generated, with a split of 100 classes as well for training, validation and testing respectively.

ResNet-12 has been used as our backbone to be consistent with previous methods by [157, 188, 229]. The ResNet-12 architecture contains 4 residual blocks with 64, 160, 320, 640 filters, each with 3×3 convolutions. We apply a 2×2 max-pooling operation after the first 3 blocks, and also a global average pooling after the last block. For optimisation, we use SGD with an initial learning rate of 0.05, momentum of 0.9, and a weight decay of $5e{-}4$. The learning rate is reduced after epoch 60 by a factor of 0.1. Following [188, 199], the model is trained for 65 epochs on both datasets, with augmentations [14, 204]. Further, the hyper-parameters α, β are tuned on a validation set, and we use the same value of 4.0 as in [230] for temperature coefficient T during knowledge distillation.

6.6 FSL Classification Results

Our results are shown in Table 6.1 and Table 6.2. The results at Stage One indicate the state-of-the-art (SOTA) performance on MiniImageNet by approximately 1% on both 5-shots and 10-shot classification tasks. The same results can be observed on the CIFAR-FS dataset. Our method shows an improvement of between 0.71% and 20% percentage points on 5-way 5-shot, and at least 3 percentage points on 5-way 10-shot learning implemented in this work. Some large percentage margins are due to the backbones used, and the number of shots. The same trend can be observed on the CIFAR-FS dataset with consistent percentage gains after Stage One. The percentage gain of our model during Stage One is

largely attributed to the use of self-supervision which enables the model to learn a more diverse and generalizable embedding space.

Stage Two that incorporates knowledge distillation produces even better results compared to Stage One. On MiniImageNet, we achieve 78.93% and 86.54% on 5-shots and 10-shots, respectively. These are gains of 6.7% and 9.0%, for instance, on ProtoNet and RelationNet, respectively. Similar consistent gains of 2-10% over SOTA results can be observed on the CIFAR-FS dataset. Other few-shot models that use knowledge distillation, for instance, RFS-distill [230] uses up to four generations for model distillation, while our model only enforces only a single generation. Our model's performance can be attributed to the way we use knowledge distillation to enforce changes in the embedding space and at the same time minimizing the representation distance between the input images and their corresponding augmentations pairs, thereby enhancing the representation capabilities of the model.

Using only the cross-entropy loss during Stage One achieves 79.64% and 72.94% on 5-shot task on CIFAR-FS and MiniImageNet, respectively. With self-supervision, including augmentations, the model performance improves to 86.40% and 78.43%, indicating significant gains. The results in Table 6.1 indicate that self-supervision at Stage One contributes for the performance improvement on Stage Two. Furthermore, during Stage Two, the advantage of using the $\mathcal{L}_{l_2}$ loss to bring logits of the augmentations closer, is demonstrated in Table 6.1. We can see that, even for both Stage One models trained on $\mathcal{L}_{ce}$ and $\mathcal{L}_{ce}$ + $\alpha\mathcal{L}_{ss}$, addition of $\mathcal{L}_{l_2}$ loss during Stage Two gives about $\sim 1\%$ gain compared with using knowledge distillation only. Varying the contribution of self-supervision over classification α during Stage One does not change the performance of the model, thus pointing to the importance of self-supervision. We observe a similar trend as for the case of α, that the performance first improves for $0 \leqslant \beta \leqslant 0.1$, and then decreases with larger values of β. However, even if we change β from 0.1 to 0.5, the performance drops only by a small margin. These results comprehensively establish individual importance of self-

supervision, knowledge distillation and ensuring the proximity of augmented versions of the image into the output space.

Table 6.1: Comparisons of few-shot learning results on MiniImageNet using various models and backbones, and our model. ResNet backbones load the pre-trained weights from ImageNet. Some results courtesy of [188, 213]. * indicates re-implementation with ResNet-12.

Method	**Backbone**	**5-Shot**	**10-Shot**
ProtoNet [239]	Conv4	51.65	
ProtoNet*	ResNet-12	72.0	77.20
MAML [60]	Conv4	53.52	
MAML*	ResNet-12	71.5	77.94
TAML [98]	ResNet-12	57.52	77.94
RelationNet [224]	Conv4	58.10	
RelationNet*	Resnet-12	69.3	79.87
SNAIL [157]	Resnet-12	68.88	
RFS-distill [230]	Resnet-12	64.82	82.14
TADAM [172]	Resnet-12	58.50	76.70
Ours-Stage one	Resnet-12	72.94	82.12
Ours-Stage Two	Resnet-12	**78.43**	**84.73**

6.7 Conclusion and Future Work

In this chapter, we explore the capability of contrastive learning via self-supervised learning to improve the metric representation capacity of few-shot learning models. Our approach operates in two stages: First, the model learns to classify inputs such that the

Table 6.2: Few-shot learning results on CIFAR-FS using various models and backbones, and our model. ResNet backbones load the pre-trained weights from ImageNet. Some results courtesy of [188, 213]. * indicates re-implementation with ResNet-12.

Method	**Backbone**	**5-Shot**	**10-Shot**
ProtoNet	Conv4	55.50	71.50
ProtoNet*	ResNet-12	57.17	79.18
MAML	Conv4	58.90	71.50
MAML*	ResNet-12	63.59	79.78
RelationNet	Conv4	58.10	72.55
RelationNet*	Resnet-12	53.10	79.87
SNAIL	Resnet-12	53.10	79.87
RFS-distill	Resnet-12	73.90	86.90
TADAM*	Resnet-12	66.72	84.41
Ours-Stage One	Resnet-12	79.64	84.5
Ours-Stage Two	Resnet-12	**86.40**	**89.10**

diversity in the outputs is maintained to avoid over-fitting and modelling the natural output manifold structure. Once learned, our approach trains a student model that preserves the original output manifold structure while jointly maximizing the model's ability to differentiate between the learned embeddings. Our empirical results on MiniImageNet and CIFAR-FS provides some insights, 1) equivariant representations result in retaining features that work well for novel classes, and 2) self-supervised learning learns general features that might be useful for other downstream tasks such as image classification and instance image segmentation that may require expensive annotations, and 3) equivariant representations perform quite well in few-shot settings where data samples are scarce. The results in this chapter show the benefit of using self-supervised learning where it

Table 6.3: Few-shot classification results on MiniImageNet and CIFAR-FS with different loss functions for Stage One and Stage Two.

		MiniImageNet		CIFAR-FS	
Stage	**Loss Function**	**5-Shot**	**10-Shot**	**5-Shot**	**10-Shot**
Stage One	$\mathcal{L}_{ce}$	79.64	84.73	86.40	89.1
	$\mathcal{L}_{ce}$ + $\alpha\mathcal{L}_{ss}$	85.15	84.18	88.01	91.97
Stage Two	$\mathcal{L}_{ce} \rightarrow \mathcal{L}_{KD}$	82.14	83.76	86.91	92.35
	$\mathcal{L}_{ce} \rightarrow \mathcal{L}_{KD} + \beta\mathcal{L}_{l_2}$	81.84	83.35	87.64	89.21
	$\mathcal{L}_{ce} + \alpha\mathcal{L}_{ss} \rightarrow \mathcal{L}_{KD}$	83.64	86.57	88.73	88.78
	$\mathcal{L}_{ce} + \alpha\mathcal{L}_{ss} \rightarrow \mathcal{L}_{KD} + \beta\mathcal{L}_{l_2}$	83.54	87.21	88.91	90.19

compares favourably with the state-of-the-art for few-shot image classification.

The following chapter provides a summary of novel contributions, and a summary of this thesis.

Chapter 7

Conclusion

7.1 Summary and Novel Contributions

The success of deep learning models has largely been attributed to the availability of large-scale datasets. However, the acquisition of large amounts of labelled datasets is infeasible in several real world problems due to the costs of annotations and rarity of some events. Inspired by how humans learn, FSL targets this problem by learning a model on a set of base classes and attempts to adapt this to novel classes using limited amounts of data. Humans, and other animals can learn a new concept from just one example by using previous knowledge that they already have [57].

FSL has predominantly been solved using optimisation-based meta-learning and metric learning [61, 190, 206, 208, 214, 239]. These approaches explained in Chapter 2, generally train a base learner that can learn a model using a few examples. Many other follow-up studies to improve these predominant methods have been explored, making FSL in image processing a growing area of research in recent years. Strong prior knowledge and experience [20, 28, 42, 62, 71, 221, 231, 241] play a significant role in this discrepancy between animal and deep neural networks. Similarity learning, therefore turns out to be a very important strategy in FSL.

Throughout this thesis, we proposed several novel deep learning-based methods specifically designed to solve problems in which image data is scarce. In particular, we considered four computer vision problems of image classification, object detection, panoptic segmentation, and self-supervised knowledge distillation in FSL. We believe that several research directions can arise from this thesis, and particularly from this area of FSL in computer vision. It is an area in its infancy, with vast, foreseeable applications in deep learning systems development. The area is developing rapidly, and its applications are becoming a staple in our daily lives in homes, education, entertainment and industry. It have given the computers the power to imagine and create new artefacts, and their development continues everyday.

In Chapter 3, we introduce a novel meta-learning model for few-shot classification that consists of dual meta-learners supervised by a central controller responsible for the control of feature extraction, meta-learning, and meta-ensemble module for integrated inference and generalisation. Our method is competitive with other state-of-the-art few-shot classifiers that make use of meta-learners, and those that employ metric-learners. Each meta-learner is composed of a pre-trained encoder fine-tuned by batch training and a parameter-free decoder used for prediction. We use ResNet-152 for learning feature representations f_θ of input and ImageNet pre-trained weights. We then optimize the classifier by using the cosine distance with a learnable scale parameter in the feature space in the meta-training stage. We provide empirical evaluation on the Omniglot, Oxford Flowers102 and MiniImageNet datasets. Our approach differs from the widely adopted meta-learning approaches due to the introduction of the dual meta-learners, and the implementation of the central controller. This area of meta-learning in few-shot classification is still being widely studied.

A recent wave of new object detection benchmarks [16, 20, 21, 130, 219, 234, 241, 243] aims at addressing the recognition of object detection inherent in these systems, coupled with identifying the objects' specific positions in images. The use of the Transformer,

based on the attention mechanism is widely accredited for improving these object detection applications. Chapter 4 introduces an approach for few-shot object detection that meta-learns object localisation and classification in an end-to-end manner based on the Transformer. Our method, based on DETR [20], encodes input images into feature embeddings that then feed into a category-agnostic decoder to generate predictions for the specific object categories. To facilitate meta-learning, a module is designed that aligns semantics of high-level and low-level features representations. All the modules are designed in multi-scale architecture to enable multi-scale object detection. By leveraging these diverse few-shot models, coupled with new few-shot learning datasets, and adopting benchmarks with stricter evaluation protocols, we believe that the object detection and vision community will make significant leaps forward.

Chapter 5 contributes to the problem of object recognition and detection, and also adds image segmentation to instance objects and the background in panoptic segmentation by producing a mask around objects and the background. The chapter presents a model for scene understanding using few-shot end-to-end panoptic segmentation model that aims to predict and represent instance objects and background regions in a fully-convolutional backbone. The approach, based on MaxDeepLab [243] first extracts multiple features in a shared decoder-encoder network, and uses an object detector from support examples capable of separating target objects from the background thereby resolving class overlaps for non-overlapping segmentation using masks. Due to the difficulty associated with combining the two separate tasks of instance segmentation and semantic segmentation, panoptic segmentation has not been widely studied though it represents one of the areas in computer vision with various application areas.

Cognisant of the fact that deep learning models can easily over-fit on the scarce data available in FSL settings, Chapter 6 takes a different approach by proposing to learn robust equivariant representations of image inputs, and corresponding true output classification manifold via self-supervised learning and knowledge distillation for few-shot

image classification. Our approach operates in two stages, first, the model learns to classify inputs such that the diversity in the outputs is not lost in augmented versions, thereby avoiding over-fitting and modelling the natural output manifold structure. Once this structure is learned, our approach uses a teacher model that trains a student model through knowledge distillation to enhance generalizability for image classification.

In this thesis, we adopted problem setups of classification, object detection, and panoptic segmentation in few-shot learning using different datasets that have been designed for different purposes, including for classification, object detection and panoptic segmentation. We focussed on proposing novel techniques in few-shot learning settings, and left considerations of domain differences and datasets to future work. These datasets exhibit very distinctive biases as shown by the differences in results, the reason why different datasets have often been considered as being entirely different domains. At the moment, working on universal representation learning from multiple domains, e.g. by [130, 186, 241, 287] remains a challenge. It would be valuable to understand if, in a more challenging scenario in which domains are disjoint, meta-learning methods have significant edge over similarity-based approaches such as prototypical networks.

7.2 Broader Impact

This work aims to equip computers systems with capabilities to learn new concepts using only a few examples, an area known as few-shot learning. We believe that several research directions can arise from this thesis, and particularly from this area of few-shot learning in computer vision. Developing deep learning models that can generalize to a large number of object classes using only a few examples is a challenging problem area, especially with limited computing resources. The area has numerous potential applications with a positive impact on society. Examples include robotics, augmented reality and virtual reality, among many others, and can be used in many areas such as manufactur-

ing, health care, agriculture, transportation and sports. Vision systems allow computer systems to understand the environment around them and enhance the capabilities of machines to see, recognise, and identify instances the same way people do. Few-shot learning takes machines closer to humans by learning from few examples, with the potential to reduce expensive and laborious data acquisition and expensive annotation effort required to learn models in domains including image classification, object detection, and panoptic segmentation.

The approaches used in this thesis compare favourably with the state-of-the-art for few-shot learning on selected datasets in terms of performance. Many novel computer vision datasets and algorithms are being developed, and they incorporate such parameters as ten-fold cross validation; ideal model features, model parameters, and learning curves; loss functions and metrics; other metrics for classification such as cross-entropy, precision, recall, f1 Score, and AUC ROC. These developments together with the availability of free, special-purpose accelerators such as GPUs and TPUs by big enterprises such as Google have accelerated the processing of deep learning workloads, and the availability of these model comparison parameters, and they have been used in this work. Other parameters such as the ease of training, speed of processing, development-based parameters such as the use of statistical tests, for example ANOVA, Chi-Square; model lifetime, or production-based parameters such as time and space complexity, offline and online learning remain as future work.

UNISA university of south africa

UNISA COLLEGE OF SCIENCE, ENGINEERING AND TECHNOLOGY'S (CSET) RESEARCH AND ETHICS COMMITTEE

18 September 2017

Ref #: 063/EZ/2017/CSET_SOC
Name: Edward Zimundzi
Student #: 51501813

Dear Mr. Edward Zimundzi

Decision: Ethics Approval for 5 years
(No humans involved)

Researcher: Edward Zimundzi
University of Botswana, DMSE, Private Bag UB00702, Gaborone, Botswana
51501813@mylife.unisa.ac.za, + 00 267 355 2063

Supervisor (s): Prof C. Omlin, omlinc@unisa.c.za,
Prof I. Sanders, sandeid@unisa.ac.za, +27 11 471 2858

Proposal: Surveillance, Identification and Monitoring of Coastal Mangrove vegetation from High Resolution Multispectral Images using Drones

Qualification: PhD Computer Science

Thank you for the application for research ethics clearance by the Unisa College of Science, Engineering and Technology's (CSET) Research and Ethics Committee for the above mentioned research. Ethics approval is granted for a period of five years from 18 September 2017 to 18 September 2022.

1. The researcher will ensure that the research project adheres to the values and principles expressed in the UNISA Policy on Research Ethics.
2. Any adverse circumstance arising in the undertaking of the research project that is relevant to the ethicality of the study, as well as changes in the methodology, should be communicated in writing to the Unisa College of Science, Engineering and

Technology's (CSET) Research and Ethics Committee. An amended application could be requested if there are substantial changes from the existing proposal, especially if those changes affect any of the study-related risks for the research participants.

3. The researcher will ensure that the research project adheres to any applicable national legislation, professional codes of conduct, institutional guidelines and scientific standards relevant to the specific field of study.
4. Only de-identified research data may be used for secondary research purposes in future on condition that the research objectives are similar to those of the original research. Secondary use of identifiable human research data require additional ethics clearance.

Note:

The reference number 063/EZ/2017/CSET_SOC should be clearly indicated on all forms of communication with the intended research participants, as well as with the Unisa College of Science, Engineering and Technology's (CSET) Research and Ethics Committee

Yours sincerely

Adèle da Veiga

Dr. A Da Veiga

Chair: Ethics Sub-Committee School of Computing, CSET

Prof I. Osunmakinde

Director: School of Computing, CSET

(PROF FW ADENTON)

Prof B. Mamba

Executive Dean: College of Science, Engineering and Technology (CSET)

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

References

[1] Arman Afrasiyabi, Jean-Francois Lalonde, and Christian Gagne. Associative Alignment for Few-shot Image Classification. In *European Conference on Computer Vision (ECCV 2020) Online*, pages 1–17, 2020.

[2] Amit Alfassy, Leonid Karlinsky, Amit Aides, Joseph Shtok, Sivan Harary, Rogerio Feris, Raja Giryes, and Alex M. Bronstein. LaSO: Label-set operations networks for multi-label few-shot learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, pages 6548–6557, Long Beach, CA, USA, 2019. IEEE. URL `https://ieeexplore.ieee.org/document/8954088`.

[3] Rahaf Aljundi, Klaas Kelchtermans, and Tinne Tuytelaars. Task-Free Continual Learning. In *Computer Vision and Pattern Recognition (CVPR 2019)*, pages 11254–11263, 2019.

[4] A. L Amutha, Annie R. Uthra, Preetha J. Roselyn, and Golda R. Brunet. Streaming data classification using hybrid classifiers to tackle stability-plasticity dilemma and concept drift. *The IEEE International Conference on Learning Representations (ICLR 2020)*, 6(i):1–19, 2019. URL `http://arxiv.org/abs/1911.09514`.

[5] Marcin Andrychowicz, Misha Denil, Sergio Gómez Colmenarejo, Matthew W. Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. In *30th Conference on*

Advances in Neural Information Processing Systems, pages 3988–3996, Barcelona, Spain, 2016.

[6] Antreas Antoniou, Amos Storkey, and Harrison Edwards. How to train your MAML. In *7th International Conference on Learning Representations, ICLR 2019*, pages 1–11, 2019.

[7] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017.

[8] Haoli Bai, Jiaxiang Wu, Irwin King, and Michael Lyu. Few shot network compression via cross distillation. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, pages 3203–3210, 2020. ISSN 2159-5399. doi: 10.1609/aaai.v34i04.5718. URL `http://arxiv.org/abs/1911.09450`.

[9] Ankan Bansal, Karan Sikka, Gaurav Sharma, and Rama Chellappa. Zero-Shot Object Detection. In *Proceedings of the European Conference on Computer Vision (ECCV 2018)*, pages 384–400, 2018.

[10] Peyman Bateni, Jan Willem van de Meent, Jarred Barber, and Frank Wood. Improving Few-Shot Visual Classification with Unlabelled Examples. *arXiv preprint arXiv: 2006.12245*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2006.12245`.

[11] Irwan Bello, Barret Zoph, Quoc Le, Ashish Vaswani, and Jonathon Shlens. Attention augmented convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2019-Octob, pages 3285–3294, 2019. ISBN 9781728148038. doi: 10.1109/ICCV.2019.00338.

[12] Nihar Bendre, Hugo Terashima Marin, and Peyman Najafrad. Learning from very

few samples: A survey. *arXiv preprint arXiv: 2007.15484*, pages 1–17, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2009.02653`.

[13] Etienne Bennequin. Meta-learning algorithms for few-shot computer vision. Technical report, Institut Polytechnique de Paris, Paris, 2019.

[14] Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H.S. Torr. Fully-convolutional Siamese networks for object tracking. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9914 LNCS:850–865, 2016. ISSN 16113349. doi: 10.1007/978-3-319-48881-3-56.

[15] Luca Bertinetto, João Henriques, Philip H.S. Torr, Andrea Vedaldi, João Henriques, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *7th International Conference on Learning Representations, ICLR 2019*, pages 1–15, 2019. ISSN 23318422.

[16] Alexey Bochkovskiy, Chien Yao Wang, and Hong Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv preprint arXiv: 2004.10934*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2004.10934`.

[17] Ujwal Bonde, Pablo F. Alcantarilla, and Stefan Leutenegger. Towards bounding-box free panoptic segmentation. *arXiv preprint arXiv: 2002.07705*, 2020. ISSN 23318422. doi: 10.1007/978-3-030-71278-5-23. URL `http://arxiv.org/abs/2002.07705`.

[18] Cristian Bucila, Rich Caruana, and Alexandru Niculescu-Mizil. Model Compression. *The International Conference on Knowledge Discovery and Data Mining (KDD2006)*, 54(1):1–9, 2006. ISSN 0218-0014.

[19] Kaidi Cao, Maria Brbic, and Jure Leskovec. Concept Learners for

Few-Shot Learning. *arXiv preprint arXiv: 2007.07375*, 2020. URL http://arxiv.org/abs/2007.07375.

[20] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers. *arXiv preprint arXiv: 2005.12872*, 12346 LNCS:213–229, 2020. ISSN 16113349. doi: 10.1007/978-3-030-58452-8-13. URL http://arxiv.org/abs/2005.12872.

[21] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. *arXiv preprint arXiv: 2104.14294*, 2021. URL http://arxiv.org/abs/2104.14294.

[22] Da Chen, Yuefeng Chen, Yuhong Li, Feng Mao, Yuan He, and Hui Xue. Self-Supervised Learning for Few-Shot Image Classification. *arXiv preprint arXiv: 1911.06045*, pages 1745–1749, 2021. ISSN 23318422. doi: 10.1109/icassp39728.2021.9413783. URL http://arxiv.org/abs/1911.06045.

[23] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. *Advances in Neural Information Processing Systems*, pages 743–752, 2017. ISSN 10495258.

[24] Hao Chen, Yali Wang, Guoyou Wang, and Yu Qiao. LSTD : A Low-Shot Transfer Detector for Object Detection, 2018. URL http://arxiv.org/abs/1803.01529.

[25] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Adam Hartwig. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *European Conference on Computer Vision (ECCV 2018)*, 34 (1), 2018. ISSN 15113701.

[26] Mark Chen, Alec Radford, Rewon Child, Jeff Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *37th International Conference on Machine Learning, ICML 2020*, volume PartF16814, pages 1669–1681, 2020. ISBN 9781713821120.

[27] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *37th International Conference on Machine Learning, ICML 2020*, PartF16814(Figure 1):1575–1585, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2002.05709`.

[28] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Y-C Wang, and J-B Huang. A Closer Look at Few-shot Classification. In *Computer Vision and Pattern Recognition (ICLR 2019)*, number 2018, pages 1–16, 2019.

[29] Weijie Chen, Shiliang Pu, Di Xie, Shicai Yang, Yilu Guo, and Luojun Lin. Unsupervised image classification for deep representation learning. *arXiv preprint arXiv: 2006.11480*, pages 1–16, 2020. ISSN 23318422. doi: 10.1007/978-3-030-66096-3-30. URL `http://arxiv.org/abs/2006.11480`.

[30] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A New Meta-Baseline for Few-Shot Learning. *arXiv preprint arXiv: 2003.04390*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2003.04390`.

[31] Yudong Chen, Chaoyu Guan, Zhikun Wei, Xin Wang, and Wenwu Zhu. MetaDelta: A Meta-Learning System for Few-shot Image Classification. *arXiv preprint arXiv: 2012.10744*, 2021. URL `http://arxiv.org/abs/2102.10744`.

[32] Zitian Chen, Yanwei Fu, Yinda Zhang, Yu Gang Jiang, Xiangyang Xue, and Leonid Sigal. Multi-Level Semantic Feature Augmentation for One-Shot Learning. *IEEE Transactions on Image Processing*, 28(9):4594–4605, 2019. ISSN 19410042. doi: 10.1109/TIP.2019.2910052.

[33] Zitian Chen, Yu-xiong Wang, and Lin Ma. Image Deformation Meta-Networks for One-Shot Learning. In *Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, volume 1, pages 8680–8689, 2019.

[34] Bowen Cheng, Maxwell D. Collins, Yukun Zhu, Ting Liu, Thomas S. Huang, Hartwig Adam, and Liang Chieh Chen. Panoptic-DeepLab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12475–12485, CVPR Virtual, 2020.

[35] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-Pixel Classification is Not All You Need for Semantic Segmentation. *arXiv preprint arXiv:2107.06278*, 2021. URL `http://arxiv.org/abs/2107.06278`.

[36] Yu Cheng, Mo Yu, Xiaoxiao Guo, and Bowen Zhou. Few-shot learning with meta metric learners. *arXiv preprint arXiv:1991.09890*, (Nips):1–9, 2019. ISSN 23318422. URL `http://arxiv.org/abs/1901.09890`.

[37] Wen Hsuan Chu, Yu Jhe Li, Jing Cheng Chang, and Yu Chiang Frank Wang. Spot and learn: A maximum-entropy patch sampler for few-shot image classification. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:6244–6253, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00641.

[38] Ching Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. Debiased Contrastive Learning. *arXiv preprint arXiv: 2007.00224*, (NeurIPS):1–20, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2007.00224`.

[39] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes Dataset

for Semantic Urban Scene Understanding. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016. doi: 10.1080/17843286.1974.11735726.

[40] Cheng Cui, Ruoyu Guo, Yuning Du, Dongliang He, Fu Li, Zewu Wu, Qiwen Liu, Shilei Wen, Jizhou Huang, Xiaoguang Hu, Dianhai Yu, Errui Ding, and Yanjun Ma. Beyond Self-Supervision: A Simple Yet Effective Network Distillation Alternative to Improve Backbones. *arXiv preprint arXiv: 2103.05959*, pages 1–10, 2021. URL `http://arxiv.org/abs/2103.05959`.

[41] Daan De Geus, Panagiotis Meletis, and Gijs Dubbelman. Panoptic segmentation with a joint semantic and instance segmentation network. *arXiv preprint arXiv: 1809.02110*, 2018. ISSN 23318422. URL `http://arxiv.org/abs/1809.02110`.

[42] Daan De Geus, Panagiotis Meletis, and Gijs Dubbelman. Single network panoptic segmentation for street scene understanding. In *IEEE Intelligent Vehicles Symposium, Proceedings*, volume 2019-June, pages 709–715, Eindhoven, The Netherlands, 2019. ISBN 9781728105604. doi: 10.1109/IVS.2019.8813788.

[43] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (May 2014):248–255, 2010. doi: 10.1109/cvpr.2009.5206848.

[44] Jingyu Deng, Xiang Li, and Yi Fang. Few-shot object detection on remote sensing images. *arXiv preprint arXiv: 2006.07826*, pages 1–12, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2006.07826`.

[45] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*

HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, volume 1, pages 4171–4186, 2019. ISBN 9781950737130.

[46] Guneet S. Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. In *The International Conference on Learning Representations (ICLR 2020)*, pages 1–20, 2019.

[47] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2015 Inter, pages 1422–1430, 2016. ISBN 9781467383912. doi: 10.1109/ICCV.2015.167.

[48] Carl Doersch, Ankush Gupta, and Andrew Zisserman. CrossTransformers : spatially-aware few-shot transfer. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, number NeurIPS, pages 1–13, Vancouver, Canada, 2020.

[49] Jeff Donahue, Trevor Darrell, and Philipp Krähenbühl. Adversarial feature learning. In *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, number 2016, pages 1–18, 2017.

[50] Xingping Dong and Jianbing Shen. Triplet loss in siamese network for object tracking. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11217 LNCS, pages 472–488, 2018. ISBN 9783030012601. doi: 10.1007/978-3-030-01261-8-28.

[51] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is

Worth 16 x 16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICVP 2021)*, 2021. URL `http://arxiv.org/abs/2010.11929`.

[52] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *International Conference on Computational Vision (ICCV 2020)*, pages 3723–3731, Venice, Italy, 2019.

[53] Thomas Elsken, Benedikt Staffler, Jan Hendrik Metzen, and Frank Hutter. Meta-Learning of Neural Architectures for Few-Shot Learning. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 12362–12372, 2020. doi: 10.1109/CVPR42600.2020.01238.

[54] Qi Fan, Wei Zhuo, Chi Keung Tang, and Yu Wing Tai. Few-Shot Object Detection with Attention-RPN and Multi-Relation Detector. In *Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, pages 4013–4022, 2019.

[55] Zhibo Fan, Jin Gang Yu, Zhihao Liang, Jiarong Ou, Changxin Gao, Gui Song Xia, and Yuanqing Li. FGN: Fully guided network for few-shot instance segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 9169–9178, 2020. ISSN 10636919. doi: 10.1109/CVPR42600.2020.00919.

[56] Li Fei-fei, Rob Fergus, and Pietro Perona. A Bayesian Approach to Unsupervised One-Shot Learning of Object Categories. In *Proceedings of the IEEE International Conference on Computer Vision 2003*, pages 0–7, 2003. ISBN 0769519504.

[57] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, 2006. ISSN 01628828. doi: 10.1109/TPAMI.2006.79.

[58] Michael Fink. Object classification from a single example utilizing class relevance metrics. In *Advances in Neural Information Processing Systems*, pages 449–456, 2005. ISBN 0262195348.

[59] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *Advances in Neural Information Processing Systems*, number Nips, pages 64–72, 2016.

[60] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *Proceedings of the 34 th International Conference on Machine Learning*, volume 3, pages 1856–1868, Sydney, Australia, 2017. ISBN 9781510855144.

[61] Chelsea Finn, Tianhe Yu, Tianhao Zhang, Pieter Abbeel, and Sergey Levine. One-shot visual imitation learning via meta-learning. In *Proceedings of Machine Learning Research*, number CoRL, pages 357–368, 2017.

[62] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic Model-Agnostic Meta-Learning. In *32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*, number NeurIPS, Montréal, Canada, 2018.

[63] Sebastian Flennerhag, Andrei A. Rusu, Razvan Pascanu, Francesco Visin, Hujun Yin, and Raia Hadsell. Meta-Learning with Warped Gradient Descent. In *3rd Workshop on Meta-Learning at NeurIPS 2019*, Vancouver, Canada, 2019. URL `http://arxiv.org/abs/1909.00025`.

[64] Ahmed Frikha, Hans Georg Köpken, Denis Krompaß, and Volker Tresp. Few-Shot One-Class Classification via Meta-Learning. *arXiv preprint arXiv: 2007.04146*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2007.04146`.

[65] Cheng Yang Fu, Tamara Berg, and Alexander Berg. IMP: Instance mask projection for high accuracy semantic segmentation of things. *Proceedings of the IEEE In-*

ternational Conference on Computer Vision, 2019-Octob:5177–5186, 2019. ISSN 15505499. doi: 10.1109/ICCV.2019.00528.

[66] Kun Fu, Tengfei Zhang, Yue Zhang, Menglong Yan, Zhonghan Chang, Zhengyuan Zhang, and Xian Sun. Meta-SSD: Towards Fast Adaptation for Few-Shot Object Detection with Meta-Learning. *IEEE Access*, 7:77597–77606, 2019. ISSN 21693536. doi: 10.1109/ACCESS.2019.2922438.

[67] Siddhartha Gairola, Mayur Hemani, Ayush Chopra, and Balaji Krishnamurthy. SimPropNet: Improved similarity propagation for few-shot image segmentation. In *IJCAI International Joint Conference on Artificial Intelligence*, volume 2021-Janua, pages 573–579, 2020. ISBN 9780999241165. doi: 10.24963/ijcai.2020/80.

[68] Hang Gao, Zheng Shou, Alireza Zareian, Hanwang Zhang, and Shih Fu Chang. Low-shot learning via covariance-preserving adversarial augmentation networks. In *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*, number NeurIPS, pages 1–11, 2018.

[69] Naiyu Gao, Yanhu Shan, Xin Zhao, and Kaiqi Huang. Learning category- and instance-aware pixel embedding for fast panoptic segmentation, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2009.13342`.

[70] Weifeng Ge, Weilin Huang, Dengke Dong, and Matthew R. Scott. Deep Metric Learning with Hierarchical Triplet Loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[71] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Prez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. In *Proceedings of the IEEE International Conference on Computer Vision 2019*, pages 8059–8068, 2019.

[72] Spyros Gidaris, Nikos Komodakis, and Ponts Paristech. Generating Classification Weights with GNN Denoising Autoencoders for Few-Shot Learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019.

[73] Ross Girshick. Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015. ISSN 15505499. doi: 10.1109/ICCV.2015.169.

[74] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. ISSN 10636919. doi: 10.1109/CVPR.2014.81.

[75] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. ISSN 15577317. doi: 10.1145/3422622.

[76] Jianping Gou, Baosheng Yu, Stephen J. Maybank, and Dacheng Tao. Knowledge Distillation: A Survey. *International Journal of Computer Vision*, 129(6): 1789–1819, 2021. ISSN 15731405. doi: 10.1007/s11263-021-01453-z. URL `http://arxiv.org/abs/2006.05525`.

[77] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural Turing Machines. *arXiv preprint arXiv: 1410.5401*, pages 1–26, 2014. URL `http://arxiv.org/abs/1410.5401`.

[78] Jean Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu,

Rémi Munos, and Michal Valko. Bootstrap Your Own Latent A New Approach to Self-Supervised Learning. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, number NeurIPS, pages 1–14, Vancouver, Canada, 2020.

[79] Yiluan Guo and Ngai Man Cheung. Attentive weights generation for few shot learning via information maximization. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 13496–13505, 2020. doi: 10.1109/CVPR42600.2020.01351.

[80] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2019-June, pages 5351–5359, 2019. ISBN 9781728132938. doi: 10.1109/CVPR.2019.00550.

[81] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1735–1742, 2006. ISBN 0769525970. doi: 10.1109/CVPR.2006.100.

[82] Bharath Hariharan and Ross Girshick. Low-Shot Visual Recognition by Shrinking and Hallucinating Features. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2017-Octob, pages 3037–3046, 2017. ISBN 9781538610329. doi: 10.1109/ICCV.2017.328.

[83] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2016-Decem, pages 770–778, 2016. ISBN 9781467388504. doi: 10.1109/CVPR.2016.90.

[84] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *2017 IEEE International Conference on Computer Vision (ICCV)*,

pages 386–397, Venice, Italy, 2017. doi: 10.1109/ICCV.2017.322. URL https://ieeexplore.ieee.org/abstract/document/8237584.

[85] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *Computer Vision and Pattern Recognition (CVPR 2020)*, pages 9729–9738, 2019. doi: 10.1109/CVPR42600.2020.00975.

[86] Olivier J. Henaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, S. M.Ali Eslami, and Aaron Vanden Oord Eslami. Data-Efficient image recognition with contrastive predictive coding. In *37th International Conference on Machine Learning, ICML 2020*, volume PartF16814, pages 4130–4140, Vienna, Austria, 2020. ISBN 9781713821120.

[87] Sean M. Hendryx, Andrew B. Leach, Paul D. Hein, and Clayton T. Morrison. Meta-Learning Initializations for Image Segmentation. *arXiv preprint arXiv: 1912.06290*, 2019. URL http://arxiv.org/abs/1912.06290.

[88] Nathan Hilliard, Lawrence Phillips, Scott Howland, Artëm Yankov, Courtney D. Corley, and Nathan O. Hodas. Few-Shot Learning with Metric-Agnostic Conditional Embeddings. *arXiv preprint arXiv: 1802.04376*, 2018. ISSN 23318422. URL http://arxiv.org/abs/1802.04376.

[89] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network. In *NIPS Deep Learning and Representation Learning Workshop (2015)*, pages 1–9, 2015. URL http://arxiv.org/abs/1503.02531.

[90] Chih-Hui Ho and Nuno Vasconcelos. Contrastive Learning with Adversarial Examples. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, number NeurIPS, Vancouver, Canada, 2020. URL http://arxiv.org/abs/2010.12050.

[91] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9370, pages 84–92, 2015. ISBN 9783319242606. doi: 10.1007/978-3-319-24261-3-7.

[92] Jonathan Huang, Chen Sun, Kevin Murphy, and Sergio Guadarrama. Speed/accuracy trade-offs for modern convolutional object detectors. In *Computer Vision and Pattern Recognition (CVPR 2020)*, pages 7310–7319, 2021.

[93] Xinyue Huo, Lingxi Xie, Xiaopeng Zhang, Longhui Wei, Hao Li, Zijie Yang, Wengang Zhou, Houqiang Li, and Qi Tian. Heterogeneous contrastive learning: Encoding spatial information for compact visual representations. *arXiv preprint arXiv: 2011.09941*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2011.09941`.

[94] Sukjun Hwang, Seoung Wug Oh, and Seon Joo Kim. Single-shot Path Integrated Panoptic Segmentation. *arXiv preprint arXiv: 2012.01632*, 1, 2020. URL `http://arxiv.org/abs/2012.01632`.

[95] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *32nd International Conference on Machine Learning, ICML 2015*, volume 1, pages 448–456, 2015. ISBN 9781510810587.

[96] Shruti Jadon. An overview of deep learning architectures in few-shot learning domain. *arXiv preprint arXiv: 2008.06365*, 2020. ISSN 23318422. doi: 10.13140/RG.2.2.31573.24803/1. URL `http://arxiv.org/abs/2008.06365`.

[97] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, Fillia Makedon, Ashwin Ramesh Babu, Debapriya Banerjee, and Fillia Make-

don. A Survey on Contrastive Self-Supervised Learning. *Technologies*, 9(1):2, 2020. ISSN 2227-7080. doi: 10.3390/technologies9010002.

[98] Muhammad Abdullah Jamal and Guo Jun Qi. Task Agnostic Meta-Learning for Few-Shot Learning. In *Computer Vision and Pattern Recognition (CVPR 2020)*, pages 11719–11727, 2020.

[99] Alexander Jaus, Kailun Yang, and Rainer Stiefelhagen. Panoramic Panoptic Segmentation: Towards Complete Surrounding Understanding via Unsupervised Contrastive Learning. *arXiv preprint arXiv:2103.00868*, 2021. URL `http://arxiv.org/abs/2103.00868`.

[100] Ziyu Jiang, Tianlong Chen, Ting Chen, and Zhangyang Wang. Robust Pre-Training by Adversarial Contrastive Learning. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, number NeurIPS, pages 1–12, Vancouver, Canada, 2020. URL `http://arxiv.org/abs/2010.13337`.

[101] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot Object Detection via Feature Reweighting. In *Computer Vision and Pattern Recognition (CVPR 2018)*, pages 8420–8429, 2018.

[102] Kai Kang, Hongsheng Li, Junjie Yan, Xingyu Zeng, Bin Yang, Tong Xiao, Cong Zhang, Zhe Wang, Ruohui Wang, Xiaogang Wang, and Wanli Ouyang. T-CNN: Tubelets with Convolutional Neural Networks for Object Detection from Videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):2896–2907, 2018. ISSN 10518215. doi: 10.1109/TCSVT.2017.2736553.

[103] Leonid Karlinsky, Joseph Shtok, Sivan Harary, Eli Schwartz, Amit Aides, Rogerio Feris, Raja Giryes, and Alex M. Bronstein. Repmet: Representative-based metric learning for classification and few-shot object detection. In *Proceedings of*

the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 2019-June, pages 5192–5201, 2019. ISBN 9781728132938. doi: 10.1109/CVPR.2019.00534.

[104] Leonid Karlinsky, Joseph Shtok, Amit Alfassy, Moshe Lichtenstein, Sivan Harary, Eli Schwartz, Sivan Doveh, Prasanna Sattigeri, Rogerio Feris, Alexander Bronstein, and Raja Giryes. StarNet: towards Weakly Supervised Few-Shot Object Detection. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)*, pages 1–13, New York, USA, 2020. URL http://arxiv.org/abs/2003.06798.

[105] Maryna Karpusha, Sunghee Yun, and Istvan Fehervari. Calibrated neighborhood aware confidence measure for deep metric learning. *arXiv preprint arXiv: 2006.04935*, 2020. ISSN 23318422. URL http://arxiv.org/abs/2006.04935.

[106] Mahmut Kaya and Hasan Şakir Bilge. Deep metric learning: A survey. *Symmetry*, 11(9), 2019. ISSN 20738994. doi: 10.3390/sym11091066.

[107] Siddhesh Khandelwal, Raghav Goyal, and Leonid Sigal. UniT: Unified Knowledge Transfer for Any-shot Object Detection and Segmentation. *arXiv preprint arXiv:2006.07502*, pages 1–19, 2020. URL http://arxiv.org/abs/2006.07502.

[108] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Dilip Krishnan, and Ce Liu. Supervised contrastive learning. *arXiv preprint arXiv: 2004.11362*, pages 1–23, 2020. ISSN 23318422. URL http://arxiv.org/abs/2004.11362.

[109] Geonuk Kim, Hong Gyu Jung, and Seong Whan Lee. Few-Shot Object Detection via Knowledge Transfer. *IEEE Transactions on Systems, Man, and*

Cybernetics: Systems, 2020-Octob:3564–3569, 2020. ISSN 21682232. doi: 10.1109/SMC42975.2020.9283497.

[110] Wonsik Kim, Bhavya Goyal, Kunal Chawla, Jungmin Lee, and Keunjoo Kwon. Attention-based ensemble for deep metric learning. In *European Conference on Computer Vision (ECCV 2018)*, pages 736–751, 2018.

[111] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollar. Panoptic feature pyramid networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:6392–6401, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00656.

[112] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollar. Panoptic segmentation. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2019-June, pages 9396–9405, 2019. ISBN 9781728132938. doi: 10.1109/CVPR.2019.00963.

[113] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese Neural Networks for One-shot Image Recognition. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, Lille, France, 2015. doi: 10.1136/bmj.2.5108.1355-c.

[114] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big Transfer (BiT): General Visual Representation Learning. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12350 LNCS: 491–507, 2020. ISSN 16113349. doi: 0.1007/978-3-030-58558-7-29.

[115] Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. CompRess: Self-Supervised Learning by Compressing Representations. In *34th Conference on Neural Information Processing Sys-*

tems (NeurIPS 2020), number NeurIPS, Vancouver, Canada, 2020. URL http://arxiv.org/abs/2010.14713.

[116] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Technical report, Technical Report TR-2009, University of Toronto,, Toronto, 2009.

[117] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6): 84–90, 2017. ISSN 15577317. doi: 10.1145/3065386.

[118] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955. ISSN 0894069X. doi: 10.1002/nav.20053. URL http://www.math.toronto.edu/mccann/1855/KuhnNRL55.pdf.

[119] S Kullback and R. A. Leibler. On Information and Sufficiency. *Annals of Mathematical Statistics*, 22(2):79–86, 2017. ISSN 2331-8422. doi: 10.24425/124266. URL http://arxiv.org/abs/1706.01538.

[120] Ravi Kumar Kushawaha, Saurabh Kumar, Biplab Banerjee, and Rajbabu Velmurugan. Distilling spikes: Knowledge distillation in spiking neural networks. *arXiv preprint arXiv: 2005.00288*, pages 1–11, 2020. ISSN 23318422. URL http://arxiv.org/abs/2005.00288.

[121] Brenden M Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua B Tenenbaum. One shot learning of simple visual concepts. In *Proceedings of the 33rd Annual Conference of the Cognitive Science Society*, volume 172, Boston, USA, 2011.

[122] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tnenbaum. Human-level concept learning through probabilistic program induction. *Cognitive Science*, 350 (6266):1332–1338, 2015. ISSN 0036-8075. doi: 10.1126/science.aab3050. URL https://www.sciencemag.org/content/350/6266/1332.full.pdf.

[123] Justin Lazarow, Kwonjoon Lee, Kunyu Shi, and Zhuowen Tu. Learning instance occlusion for panoptic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 10717–10726, 2020. ISSN 10636919. doi: 10.1109/CVPR42600.2020.01073.

[124] Yann Lecun, Le'on Bottou, Yoshua Bengio, and Parick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1998. URL `http://ieeexplore.ieee.org/document/726791/full-text-section`.

[125] Hankook Lee, Sung Ju Hwang, and Jinwoo Shin. Self-supervised Label Augmentation via Input Transformations. In *Proceedings of the 37th International Conference on Machine Learning (ICML) Online*, 2020. URL `http://arxiv.org/abs/1910.05872`.

[126] Seung Hyun Lee, Dae Ha Kim, and Byung Cheol Song. Self-supervised Knowledge Distillation Using Singular Value Decomposition. In *European Conference on Computer Vision (ECCV 2018)*, pages 1–16, 2018.

[127] Aoxue Li, Tiange Luo, Zhiwu Lu, Tao Xiang, and Liwei Wang. Large-scale few-shot learning: Knowledge transfer with class hierarchy. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2019-June, pages 7205–7213, 2019. ISBN 9781728132938. doi: 10.1109/CVPR.2019.00738.

[128] Jie Li, Allan Raventos, Arjun Bhargava, Takaaki Tagawa, and Adrien Gaidon. Learning to Fuse Things and Stuff. *arXiv preprint arXiv: 1812.01192*, 2018. ISSN 23318422. URL `http://arxiv.org/abs/1812.01192`.

[129] Shaoqi Li, Wenfeng Song, Shuai Li, Aimin Hao, and Hong Qin. Meta-RetinaNet

for Few-shot Object Detection. In *The 31st British Machine Vision (Virtual) Conference 2020*, 2020.

[130] Wei Hong Li, Xialei Liu, and Hakan Bilen. Universal Representation Learning from Multiple Domains. *arXiv preprint arXiv: 2003.13841*, 2021. URL `http://arxiv.org/abs/2103.13841`.

[131] Xiaomeng Li, Lequan Yu, Chi Wing Fu, Meng Fang, and Pheng Ann Heng. Revisiting metric learning for few-shot image classification. *Neurocomputing*, 406: 49–58, 2020. ISSN 18728286. doi: 10.1016/j.neucom.2020.04.040.

[132] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-guided unified network for panoptic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:7019–7028, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00719.

[133] Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Fully Convolutional Networks for Panoptic Segmentation. In *arXiv preprint arXiv: 2012.00720*, pages 214–223, 2020. URL `https://openaccess.thecvf.com/content/CVPR2021/papers/` `http://arxiv.org/abs/2012.00720`.

[134] Yu Li, Tao Wang, Bingyi Kang, Sheng Tang, Chunfeng Wang, Jintao Li, and Jiashi Feng. Overcoming Classifier Imbalance for Long-tail Object Detection with Balanced Group Softmax. In *Computer Vision and Pattern Recognition (CVPR 2020)*, pages 10991–11000, 2020.

[135] Yu-Jhe Li, Xinshuo Weng, and Kris M Kitani. Learning Shape Representations for Person Re-Identification under Clothing Change. In *Workshop on Applications of Computer Vision (WACV 2021)*, pages 2432–2441, 2021.

[136] Yuewen Li, Wenquan Feng, Shuchang Lyu, Qi Zhao, and Xuliang Li. MM-FSOD: Meta and metric integrated few-shot object detection. *arXiv preprint arXiv: 2012.15159*, pages 1–30, 2020. URL `http://arxiv.org/abs/2012.15159`.

[137] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-SGD: Learning to learn quickly for few-shot learning. 2017. URL `https://arxiv.org/abs/1707.098350A`.

[138] Yann Lifchitz, Sylvaine Picard, and Andrei Bursuc. Dense Classification and Implanting for Few-Shot Learning. In *Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, pages 9258–9267, 2019.

[139] Matthieu Lin, Chuming Li, Xingyuan Bu, Ming Sun, Chen Lin, Junjie Yan, Wanli Ouyang, and Zhidong Deng. DETR for Crowd Pedestrian Detection. *arXiv preprint arXiv: 2012.06785*, 2020. URL `http://arxiv.org/abs/2012.06785`.

[140] Tsung Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8693 LNCS (PART 5):740–755, 2014. ISSN 16113349. doi: 10.1007/978-3-319-10602-1-48.

[141] Tsung Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327, 2020. ISSN 19393539. doi: 10.1109/TPAMI.2018.2858826.

[142] Huanyu Liu, Chao Peng, Changqian Yu, Jingbo Wang, Xu Liu, Gang Yu, and Wei Jiang. An end-to-end network for panoptic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:6165–6174, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00633.

[143] Jinlu Liu and Yongqiang Qin. Prototype refinement network for few-shot segmentation. *arXiv preprint arXiv: 2002.03579*, 2020. ISSN 23318422. URL http://arxiv.org/abs/2002.03579.

[144] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*, 128(2):261–318, 2020. ISSN 15731405. doi: 10.1007/s11263-019-01247-4. URL https://doi.org/10.1007/s11263-019-01247-4.

[145] Liyang Liu, Bochao Wang, Zhanghui Kuang, Jing-Hao Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. GenDet: Meta Learning to Generate Detectors From Few Shots. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2021. ISSN 2162-237X. doi: 10.1109/tnnls.2021.3053005.

[146] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C. Berg. SSD: Single shot multibox detector. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9905 LNCS:21–37, 2016. ISSN 16113349. doi: 10.1007/978-3-319-46448-0-2.

[147] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. Self-supervised Learning: Generative or Contrastive. *arXiv preprint arXiv: 2006.08218*, pages 1–23, 2020. ISSN 23318422. URL http://arxiv.org/abs/2006.08218.

[148] Yongfei Liu, Xiangyi Zhang, Songyang Zhang, and Xuming He. Part-Aware Prototype Network for Few-Shot Semantic Segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture*

Notes in Bioinformatics), 12354 LNCS(18):142–158, 2020. ISSN 16113349. doi: 10.1007/978-3-030-58545-7-9.

[149] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *arXiv preprint arXiv: 2103.14030*, 2021. URL `http://arxiv.org/abs/2103.14030`.

[150] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019*, pages 847–856, 2015. ISBN 9781728150239. doi: 10.1109/ICCVW.2019.00113.

[151] Qinxuan Luo. Few-Shot Learning via Feature Hallucination with Variational Inference. In *Workshop on Applications of Computer Vision (WACV 2021) Online*, pages 3963–3972, 2021.

[152] Anay Majee, Kshitij Agrawal, and Anbumani Subramanian. Few-Shot Learning for Road Object Detection. *arXiv preprint arXiv: 2101.12543*, 2021. URL `http://arxiv.org/abs/2101.12543`.

[153] Hadi Mansourifar and Weidong Shi. One-Shot GAN Generated Fake Face Detection. *arXiv preprint arXiv: 2003.12244*, 2020. ISSN 23318422. URL `https://arxiv.org/abs/2003.12244`.

[154] Panagiotis Meletis, Xiaoxiao Wen, Chenyang Lu, Daan de Geus, and Gijs Dubbelman. Cityscapes-Panoptic-Parts and PASCAL-Panoptic-Parts datasets for Scene Understanding. *arXiv preprint arXiv: 2004.07944*, pages 1–21, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2004.07944`.

[155] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher

assistant. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, pages 5191–5198, 2020. ISSN 2159-5399. doi: 10.1609/aaai.v34i04.5963.

[156] Seyed Iman Mirzadeh, Razvan Pascanu, Mehrdad Farajtabar, and Hassan Ghasemzadeh. Understanding the Role of Training Regimes in Continual Learning. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, pages 1–20, Vancouver, Canada, 2020.

[157] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A Simple Neural Attentive Meta-Learner. *The International Conference on Learning Representations (ICLR 2018)*, 42(4):236–246, 2018. ISSN 1570145X. doi: 10.1007/s10749-008-0037-4.

[158] Rohit Mohan and Abhinav Valada. EfficientPS: Efficient Panoptic Segmentation. *International Journal of Computer Vision*, 2021. ISSN 15731405. doi: 10.1007/s11263-021-01445-z. URL `https://doi.org/10.1007/s11263-021-01445-z`.

[159] Robert C. Moore and John DeNero. L1 and L2 regularization for multiclass hinge loss models. Technical report, Google Research, 2009. URL `https://research.google/pubs/pub37362/`.

[160] Tsendsuren Munkhdalai and Hong Yu. Meta networks. In *34th International Conference on Machine Learning, ICML 2017*, volume 5, pages 3933–3943, 2017. ISBN 9781510855144.

[161] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017)*, pages 4990–4999, 2017. ISBN 9781538610329. doi: 10.1109/ICCV.2017.534.

[162] Hieu V. Nguyen and Li Bai. Cosine Similarity Metric Learning for Face Verification. In *Asian Conference on Computer Vision (ACCV 2010)*, pages 709–720, 2010. ISBN 9783642193095. doi: 10.1007/978-3-642-19309-5.

[163] Khoi Nguyen and Sinisa Todorovic. Feature weighting and boosting for few-shot segmentation. In *Proceedings of the IEEE International Conference on Computer Vision 2019*, pages 622–631, 2019.

[164] Khoi Nguyen and Sinisa Todorovic. A Self-supervised GAN for unsupervised few-shot object recognition. *arXiv preprint arXiv: 2008.06982*, 2020. ISSN 23318422. URL `https://arxiv.org/abs/2008.06982`.

[165] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv: 1803.02999*, pages 1–15, 2018. ISSN 23318422. URL `http://arxiv.org/abs/1803.02999`.

[166] Maria-elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics and Image Processing*, Bhubaneswar, India, 2008. doi: 10.1109/ICVGIP.2008.47.

[167] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1520–1528, 2015. ISBN 9781467383912. doi: 10.1109/ICCV.2015.178.

[168] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9910 LNCS:69–84, 2016. ISSN 16113349. doi: 10.1007/978-3-319-46466-4-5.

[169] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation Learning by Learning to Count. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2017-Octob, pages 5899–5907, 2017. ISBN 9781538610329. doi: 10.1109/ICCV.2017.628.

[170] Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. Boosting Self-Supervised Learning via Knowledge Transfer. In *Computer Vision and Pattern Recognition (CVPR 2020)*, pages 9359–9367, 2020. URL `http://arxiv.org/abs/`.

[171] Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. Deep Metric Learning with BIER: Boosting Independent Embeddings Robustly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):276–290, 2020. ISSN 19393539. doi: 10.1109/TPAMI.2018.2848925.

[172] Boris N. Oreshkin, Pau Rodriguez, and Alexandre Lacoste. TADAM: Task dependent adaptive metric for improved few-shot learning. In *32nd Conference on Neural Information Processing Systems (NeurIPS 2018)*, pages 1–11, Montréal, Canada, 2018.

[173] Rafael Padilla, Sergio L. Netto, and Eduardo A.B. Da Silva. A Survey on Performance Metrics for Object-Detection Algorithms. *International Conference on Systems, Signals, and Image Processing*, 2020-July:237–242, 2020. ISSN 21578702. doi: 10.1109/IWSSIP48289.2020.9145130.

[174] Yingwei Pan, Ting Yao, Yehao Li, Yu Wang, Chong Wah Ngo, and Tao Mei. Transferrable prototypical networks for unsupervised domain adaptation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:2234–2242, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00234.

[175] German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. ISSN 18792782. doi: 10.1016/j.neunet.2019.01.012.

[176] Young Hyun Park, Jun Seo, and Jaekyun Moon. CAFENet: Class-Agnostic Few-Shot Edge Detection Network. *arXiv preprint arXiv: 2003.08235*, 2020. ISSN 23318422. URL http://arxiv.org/abs/2003.08235.

[177] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, and Vincent Dubourg. Scikit-learn : Scikit-learn : Machine Learning in Python. *Journal of Machine Learning Research*, 1(2011):2825–2830, 2011. URL https://hal.inria.fr/hal-00650905v1/document.

[178] Juan-Manuel Perez-Rua, Xiatian Zhu, Timothy Hospedales, and Tao Xiang. Incremental Few-Shot Object Detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, number 2, Seattle, WA, USA, 2020. doi: 10.1109/CVPR42600.2020.01386. URL https://ieeexplore.ieee.org/document/9157715.

[179] Nguyen Huu Phong and Bernardete Ribeiro. Rethinking Recurrent Neural Networks and other improvements for image classification, 2020. ISSN 23318422. URL http://arxiv.org/abs/2007.15161.

[180] Mary Phuong and Christoph H Lampert. Towards Understanding Knowledge Distillation. In *Proceedings of the 36th International Conference on Machine Learning*, Long Beach, California, 2019.

[181] Antonio Polino, Razvan Pascanu, and Dan Alistarh. Model compression via distillation and quantization. In *The International Conference*

on Learning Representations (ICLR 2018), pages 1–21, 2018. URL https://dl.acm.org/doi/10.1145/3394171.3413832.

[182] Hang Qi, Matthew Brown, and David G Lowe. Low-Shot Learning with Imprinted Weights. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5822–5830, 2018.

[183] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan Yuille. Few-Shot Image Recognition by Predicting Parameters from Activations. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 7229–7238, 2018. ISBN 9781538664209. doi: 10.1109/CVPR.2018.00755.

[184] Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. DetectoRS: Detecting Objects with Recursive Feature Pyramid and Switchable Atrous Convolution. *arXiv preprint arXiv:2006.02334*, 2020. URL http://arxiv.org/abs/2006.02334.

[185] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, pages 1–16, 2016.

[186] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. *arXiv preprint arXiv: 2103.00020*, 2021. URL http://arxiv.org/abs/2103.00020.

[187] Shafin Rahman, Salman Khan, Nick Barnes, and Fahad Shahbaz Khan. Any-shot object detection. *arXiv preprint arXiv:2003.07003*, 2020. ISSN 23318422. URL http://arxiv.org/abs/2003.07003.

[188] Jathushan Rajasegaran, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Mubarak Shah, Salman Khan, Munawar Hayat, and Mubarak Shah. Self-supervised Knowledge Distillation for Few-shot Learning. *arXiv preprint arXiv:2006.09785*, pages 1–11, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2006.09785`.

[189] Janarthanan Rajendran. Meta-Learning Requires Meta-Augmentation. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, Vancouver, Canada, 2020.

[190] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-short learning. In *The International Conference on Learning Representations (ICLR 2017)*, volume 1, pages 1–13, 2017.

[191] Sylvestre Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. iCaRL: Incremental classifier and representation learning. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pages 5533–5542, 2017. ISBN 9781538604571. doi: 10.1109/CVPR.2017.587.

[192] Joseph Redmon and Ali Farhadi. YOLO9000: Better, faster, stronger. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua:6517–6525, 2017. doi: 10.1109/CVPR.2017.690.

[193] Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. *arXiv preprint arXiv: 1804.02767*, 2018. ISSN 23318422. URL `http://arxiv.org/abs/1804.02767`.

[194] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer So-*

ciety Conference on Computer Vision and Pattern Recognition, 2016-Decem:779–788, 2016. ISSN 10636919. doi: 10.1109/CVPR.2016.91.

[195] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-Learning for Semi-Supervised Few-Shot Classification. In *31st Conference on Neural Information Processing Systems (NIPS 2017)*, number NIPS, Long Beach, CA, USA, 2017.

[196] Mengye Ren, Renjie Liao, Ethan Fetaya, and Richard S. Zemel. Incremental few-shot learning with attention attractor networks. *Advances in Neural Information Processing Systems*, 32(NeurIPS):1–11, 2019. ISSN 10495258.

[197] Mengye Ren, Michael L. Iuzzolino, Michael C. Mozer, and Richard S. Zemel. Wandering Within a World: Online Contextualized Few-Shot Learning. *arXiv preprint arXiv: 2007.04546*, pages 1–19, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2007.04546`.

[198] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017. ISSN 01628828. doi: 10.1109/TPAMI.2016.2577031.

[199] Mamshad Nayeem Rizve. Exploring Complementary Strengths of Invariant and Equivariant Representations for Few-Shot Learning. In *Conference on Computer Vision and Pattern Recognition (CVPR 2021)*, 2021.

[200] Olaf Ronneberger, Phillip Fischer, and Thomas Brox. UNet: Convolutional Networks for Biomedical Image Segmentation. 2015. URL `http://arxiv.org/abs/1505.04597`.

[201] Vivek Roy, Kris Kitani, Yan Xu, Ruslan Salakhutdinov, Yu Xiong Wang, and Martial Hebert. Few-shot learning with intra-class knowledge transfer. *arXiv preprint arXiv: 2008.09892*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2008.09892`.

[202] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv: 1609.04747*, pages 1–14, 2016. URL `http://arxiv.org/abs/1609.04747`.

[203] Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive Neural Networks. *arXiv preprint arXiv: 1606.04671*, 2016. URL `http://arxiv.org/abs/1606.04671`.

[204] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *7th International Conference on Learning Representations, ICLR 2019*, pages 1–17, 2019. ISSN 23318422.

[205] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016.

[206] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-Learning with Memory-Augmented Neural Networks. In *33rd International Conference on Machine Learning, ICML 2016*, volume 4, pages 2740–2751, 2016. ISBN 9781510829008.

[207] Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the*

IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 07-12-June, pages 815–823, 2015. ISBN 9781467369640. doi: 10.1109/CVPR.2015.7298682.

[208] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Abhishek Kumar, Rogerio Feris, Raja Giryes, and Alex M. Bronstein. Delta-encoder: An effective sample synthesis method for few-shot object recognition. *Advances in Neural Information Processing Systems*, 32(NeurIPS 2018):2845–2855, 2018. ISSN 10495258.

[209] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. *British Machine Vision Conference 2017, BMVC 2017*, 2017. ISSN 23318422. doi: 10.5244/c.31.167.

[210] Chengchao Shen, Xinchao Wang, Youtan Yin, Jie Song, Sihui Luo, and Mingli Song. Progressive network grafting for few-shot knowledge distillation. *arXiv preprint arXiv: 2012.04915*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2012.04915`.

[211] Mennatullah Siam, Boris N. Oreshkin, and Martin Jagersand. AMP: Adaptive masked proxies for few-shot segmentation. In *Computer Vision and Pattern Recognition (ICCV 2019)*, pages 5249–5258, 2019.

[212] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–14, 2015.

[213] Jake Snell and Richard Zemel. Bayesian Few-Shot Classification with One-vs-Each Pólya-Gamma Augmented Gaussian Processes. *arXiv preprint arXiv: 2007.10417*, pages 1–34, 2020. URL `http://arxiv.org/abs/2007.10417`.

[214] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems*, 2017-Decem: 4078–4088, 2017. ISSN 10495258.

[215] Wonchul Son, Jaemin Na, and Wonjun Hwang. Densely guided knowledge distillation using multiple teacher assistants. *arXiv preprint arXiv: 2009.08825*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2009.08825`.

[216] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep Metric Learning via Lifted Structured Feature Embedding. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2016-Decem, pages 4004–4012, 2016. ISBN 9781467388504. doi: 10.1109/CVPR.2016.434.

[217] Jingwei Song, Mitesh Patel, Andreas Girgensohn, and Chelhwon Kim. Combining deep learning with geometric features for image based localization in the gastrointestinal tract. *arXiv preprint arXiv: 2005.05481*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2005.05481`.

[218] Aravind Srinivas, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Pieter Abbeel. CURL: Contrastive unsupervised representations for reinforcement learning. In *37th International Conference on Machine Learning, ICML 2020*, pages 5595–5606, 2020. ISBN 9781713821120. URL `http://arxiv.org/abs/2004.04136`.

[219] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. *arXiv preprint arXiv: 2105.05633*, pages 29–35, 2021. ISSN 01678655. doi: 10.1016/j.patrec.2021.04.024.

[220] Jong-chyi Su, Subhransu Maji, and Bharath Hariharan. When Does Self-

supervision Improve Few-shot Learning? In *European Conference on Computer Vision (ECCV 20)*, pages 1–18, 2020.

[221] Bo Sun, Banghuai Li, Shengcai Cai, Ye Yuan, and Chi Zhang. FSCE: Few-Shot Object Detection via Contrastive Proposal Encoding. *arXiv preprint arXiv: 2103.05950*, 2021. URL `http://arxiv.org/abs/2103.05950`.

[222] Qianru Sun, Yaoyao Liu, Tat Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:403–412, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00049.

[223] Qianru Sun, Yaoyao Liu, Zhaozheng Chen, Tat-seng Seng Chua, and Bernt Schiele. Meta-transfer learning through hard tasks. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–14, 2020. doi: 10.1109/tpami.2020.3018506.

[224] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H.S. Torr, and Timothy M. Hospedales. Learning to Compare: Relation Network for Few-Shot Learning. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018. ISBN 9781538664209. doi: 10.1109/CVPR.2018.00131.

[225] Christian Szegedy, Alezander Toshev, and Dumitru Erhan. Deep Neural Networks for Object Detection. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, volume 7, pages 2553–2561, 2014. doi: 10.3928/19404921-20140820-01.

[226] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going

Deeper with Convolutions. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 91, pages 2322–2330, 2015. doi: 10.1109/CVPR.2015.7298594.

[227] Ajinkya Tejankar, Soroush Abbasi Koohpayegani, Vipin Pillai, Paolo Favaro, and Hamed Pirsiavash. ISD: Self-supervised learning by iterative similarity distillation, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2012.09259`.

[228] Pinzhuo Tian, Zhangkai Wu, Lei Qi, Lei Wang, Yinghuan Shi, and Yang Gao. Differentiable meta-learning model for few-shot semantic segmentation. *AAAI 2020-34th AAAI Conference on Artificial Intelligence*, pages 12087–12094, 2020. ISSN 2159-5399. doi: 10.1609/aaai.v34i07.6887. URL `http://arxiv.org/abs/1911.10371`.

[229] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive Representation Distillation. *International Conference on Learning Representations, ICLR 2020*, (2014):1–19, 2019. ISSN 23318422. URL `http://arxiv.org/abs/1910.10699`.

[230] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. Rethinking Few-Shot Image Classification: A Good Embedding is All You Need? In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 12359 LNCS, pages 266–282, 2020. ISBN 9783030585679. doi: 10.1007/978-3-030-58568-6-16.

[231] Zhi Tian, Bowen Zhang, Hao Chen, and Chunhua Shen. Instance and Panoptic Segmentation Using Conditional Convolutions. *arXiv preprint arXiv: 2102.03026*, pages 1–12, 2021. URL `http://arxiv.org/abs/2102.03026`.

[232] Zhuotao Tian, Xin Lai, Li Jiang, Michelle Shu, Hengshuang Zhao, and Jiaya Jia.

Generalized Few-Shot Semantic Segmentation, 2020. ISSN 23318422. URL http://arxiv.org/abs/2010.05210.

[233] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers and distillation through attention. *arXiv preprint arXiv:2012.12877*, pages 1–22, 2020. URL http://arxiv.org/abs/2012.12877.

[234] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzago, and Hugo Larochelle. Meta-Dataset : A Dataset of Datasets for Learning to Learn from Few Examples. *International Conference on Learning Representations, ICLR 2020*, 2020.

[235] J. R.R. Uijlings, K. E.A. Van De Sande, T. Gevers, and A. W.M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104 (2):154–171, 2013. ISSN 09205691. doi: 10.1007/s11263-013-0620-5.

[236] Aaron Van Den Oord, Yazhe Li, Oriol Vinyals, Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding. *arXiv preprint arXiv: 1807.03748*, 2018. ISSN 23318422. URL http://arxiv.org/abs/1807.03748.

[237] Ashish Vaswani, Noam Shazeer, Niki Parnar, Jacob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *31st Conference on Neural Information Processing Systems (NIPS 2017)*, volume 8, pages 5998–6008, Long Beach, CA, USA, 2017. doi: 10.1109/2943.974352.

[238] Ricardo Vilalta and Christophe Giraud-carrier. Meta-Learning: Concepts and

Techniques. In *Data Mining and Knowledge Discovery Handbook*, pages 717–731. Springer, Boston, MA., 2009.

[239] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3637–3645, 2016.

[240] Risto Vuorio and Hexiang Hu. Multimodal Model-Agnostic Meta-Learning via Task-Aware Modulation. In *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, pages 1–12, Vancouver, Canada, 2019.

[241] Chien Yao Wang, I Hau Yeh, and Hong Yuan Mark Liao. You Only Learn One Representation: Unified Network for Multiple Tasks. *arXiv preprint arXiv: 2105.04206*, pages 1–11, 2021. URL `http://arxiv.org/abs/2105.04206`.

[242] Haochen Wang, Xudong Zhang, Yutao Hu, Yandan Yang, Xianbin Cao, and Xiantong Zhen. Few-Shot Semantic Segmentation with Democratic Attention Networks. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12358 LNCS:730–746, 2020. ISSN 16113349. doi: 10.1007/978-3-030-58601-0-43.

[243] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. MaX-DeepLab: End-to-End Panoptic Segmentation with Mask Transformers. *arXiv preprint arXiv: 2012.00759*, pages 5463–5474, 2020. URL `http://arxiv.org/abs/2012.00759`.

[244] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang Chieh Chen. Axial-DeepLab: Stand-Alone Axial-Attention for Panoptic Segmentation. *arXiv preprint arXiv: 2003.07853*, 12349 LNCS:108–126, 2020. ISSN 16113349. doi: 10.1007/978-3-030-58548-8-7. URL `http://arxiv.org/abs/2003.07853`.

[245] Shuo Wang, Jun Yue, Jianzhuang Liu, Qi Tian, and Meng Wang. Large-Scale Few-Shot Learning via Multi-modal Knowledge Discovery. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12355 LNCS:718–734, 2020. ISSN 16113349. doi: 10.1007/978-3-030-58607-2-42.

[246] Wanwei Wang, Wei Hong, Feng Wang, and Jinke Yu. GAN-Knowledge Distillation for One-Stage Object Detection. *IEEE Access*, 8:60719–60727, 2020. ISSN 21693536. doi: 10.1109/ACCESS.2020.2983174.

[247] Xin Wang, Fisher Yu, Ruth Wang, Trevor Darrell, and Joseph E. Gonzalez. TAFE-Net: Task-aware feature embeddings for low shot learning. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:1831–1840, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00193.

[248] Xin Wang, Thomas E. Huang, Trevor Darrell, Joseph E. Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. *37th International Conference on Machine Learning, ICML 2020*, PartF16814:9861–9870, 2020. ISSN 23318422.

[249] Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. Generalizing from a Few Examples: A Survey on Few-shot Learning. *ACM Computing Surveys*, 53(3):1–34, 2020. ISSN 15577341. doi: 10.1145/3386252. URL `http://arxiv.org/abs/1904.05046`.

[250] Yikai Wang, Chengming Xu, Chen Liu, Li Zhang, and Yanwei Fu. Instance Credibility Inference for Few-Shot Learning. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 12833–12842, 2020. doi: 10.1109/CVPR42600.2020.01285.

[251] Yong Wang, Xiao-Ming Wu, Qimai Li, Jiatao Gu, Wangmeng Xiang, Lei Zhang, and Victor O K Li. Large Margin Meta-Learning

for Few-Shot Classification. In *2nd Workshop on Meta-Learning at NeurIPS 2018*, pages 1–8, Montréal, Canada., 2018. URL `http://metalearning.ml/2018/papers/metalearn2018-paper11.pdf`.

[252] Yu-Xiong Wang. Meta-Learning to Detect Rare Objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9925–9934, Seoul, Korea (South), 2019.

[253] Yu Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-Shot Learning from Imaginary Data. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 7278–7286, 2018. ISSN 10636919. doi: 10.1109/CVPR.2018.00760.

[254] Mark Weber, Jonathon Luiten, and Bastian Leibe. Single-shot panoptic segmentation. *IEEE International Conference on Intelligent Robots and Systems*, pages 8476–8483, 2020. ISSN 21530866. doi: 10.1109/IROS45743.2020.9341546.

[255] Davis Wertheimer and Bharath Hariharan. Few-shot learning with localization in realistic settings. In *Computer Vision and Pattern Recognition (CVPR 2019)*, pages 6558–6567, 2019.

[256] Jiaxi Wu, Songtao Liu, Di Huang, and Yunhong Wang. Multi-Scale Positive Sample Refinement for Few-Shot Object Detection. In *European Conference on Computer Vision (ECCV 2020)*, pages 1–17, 2020.

[257] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised Feature Learning via Non-parametric Instance Discrimination. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, 2018. ISBN 9781538664209. doi: 10.1109/CVPR.2018.00393.

[258] Yang Xiao and Renaud Marlet. Few-Shot Object Detection and Viewpoint Estima-

tion for Objects in the Wild. In *European Conference on Computer Vision (ECCV 2020)*, pages 1–18, 2020.

[259] Zixuan Xiao, Ping Zhong, Yuan Quan, Xuping Yin, and Wei Xue. Few-shot object detection with feature attention highlight module in remote sensing images. In *Third International Conference on Image, Video Processing and Artificial Intelligence, 2020*, page 9, Shanghai, China, 2020. ISBN 9781510639973. doi: 10.1117/12.2577473.

[260] Shufeng Xiong, Yue Zhang, Donghong Ji, and Yinxia Lou. Distance metric learning for aspect phrase grouping. In *COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers*, pages 2492–2502, 2016. ISBN 9784879747020.

[261] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:8810–8818, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00902.

[262] Guodong Xu, Ziwei Liu, Xiaoxiao Li, and Chen Change Loy. Knowledge Distillation Meets Self-Supervision. In *European Conference on Computer Vision (ECCV 2020)*, 2020. URL `https://arxiv.org/abs/2006.07114`.

[263] Xiaopeng Yan, Ziliang Chen, Anni Xu, Xiaoxi Wang, Xiaodan Liang, and Liang Lin. Meta R-CNN : Towards General Solver for Instance-level Low-shot Learning. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), 2019.

[264] Tien Ju Yang, Maxwell D. Collins, Yukun Zhu, Jyh Jing Hwang, Ting Liu, Xiao Zhang, Vivienne Sze, George Papandreou, and Liang Chieh Chen. DeeperLab:

Single-Shot Image Parser. *arXiv preprint arXiv: 1902.05093*, 2019. ISSN 23318422. URL http://arxiv.org/abs/1902.05093.

[265] Yibo Yang, Hongyang Li, Xia Li, Qijie Zhao, Jianlong Wu, and Zhouchen Lin. SOGNet: Scene overlap graph network for panoptic segmentation. *arXiv preprint arXiv: 1911.07527*, 2019. ISSN 23318422. doi: 10.1609/aaai.v34i07.6955. URL http://arxiv.org/abs/1911.07527.

[266] Yuwei Yang, Fanman Meng, Hongliang Li, King N. Ngan, and Qingbo Wu. A new few-shot segmentation network based on class representation. *arXiv preprint arXiv: 1909,087518751*, 2019. ISSN 23318422. URL http://arxiv.org/abs/1909.08754.

[267] Han Jia Ye, Hexiang Hu, De Chuan Zhan, and Fei Sha. Few-Shot Learning via Embedding Adaptation with Set-to-Set Functions. In *Computer Vision and Pattern Recognition CVPR 2019)*, pages 8808–8817, 2020.

[268] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. A Gift from Knowledge Distillation : Fast Optimization , Network Minimization and Transfer Learning. In *Computer Vision and Pattern Recognition (CVPR 2016)*, pages 4133–4141, 2016.

[269] Junbo Yin, Wenguan Wang, Qinghao Meng, Ruigang Yang, and Jianbing Shen. A unified object motion and affinity model for online multi-object tracking. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 6767–6776, 2020. doi: 10.1109/CVPR42600.2020.00680.

[270] Xiaowen Ying, Xin Li, and Mooi Choo Chuah. Weakly-supervised Object Representation Learning for Few-shot Semantic Segmentation. In *Workshop on Applications of Computer Vision (WACV 2021)*, pages 1497–1506, 2021.

[271] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning

with dynamically expandable networks. *arXiv preprint arXiv: 1708: 01547*, pages 1–11, 2018. URL `https://arxiv.org/abs/1708.01547`.

[272] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems*, volume 2, pages 3320–3328, 2014.

[273] Zhongjie Yu, Lin Chen, Zhongwei Cheng, and Jiebo Luo. Transmatch: A transfer-learning scheme for semi-supervised few-shot learning. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 12853–12861, 2020. doi: 10.1109/CVPR42600.2020.01287.

[274] Yading Yuan, Ming Chao, and Yeh-chi Lo. Deep Fully Convolutional Networks With Jaccard Distance. *IEEE Transactions on Medical Imaging*, 36(9):1876–1886, 2017.

[275] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4L: Self-Supervised Semi-Supervised Learning. In *Computer Vision and Pattern Recognition (ICCV 2019)*, pages 1476–1485, 2019.

[276] Chi Zhang and Guosheng Lin. CANet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5217–5226, 2018.

[277] Chi Zhang, Guosheng Lin, Fayao Liu, Jiushuang Guo, Qingyao Wu, and Rui Yao. Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2019-Octob, pages 9586–9594, 2019. ISBN 9781728148038. doi: 10.1109/ICCV.2019.00968.

[278] Gongjie Zhang, Zhipeng Luo, Kaiwen Cui, and Shijian Lu. Meta-DETR: Few-Shot Object Detection via Unified Image-Level Meta-Learning, 2021. URL `http://arxiv.org/abs/2103.11731`.

[279] Hongguang Zhang, Jing Zhang, and Piotr Koniusz. Few-shot learning via saliency-guided hallucination of samples. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:2765–2774, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00288.

[280] Min Zhang, Donglin Wang, and Sibo Gai. Knowledge Distillation for Model-Agnostic. In *24th European Conference on Artificial Intelligence - ECAI 2020*, Santiago de Compostela, Spain, 2020.

[281] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9907 LNCS:649–666, 2016. ISSN 16113349. doi: 10.1007/978-3-319-46487-9-40.

[282] Richard Zhang, Phillip Isola, and Alexei A. Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, volume 2017-Janua, pages 645–654, 2017. ISBN 9781538604571. doi: 10.1109/CVPR.2017.76.

[283] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Thomas S. Huang. SG-One: Similarity Guidance Network for One-Shot Semantic Segmentation. *IEEE Transactions on Cybernetics*, 50(9):3855–3865, 2020. ISSN 21682275. doi: 10.1109/TCYB.2020.2992433.

[284] Yifei Zhang, Désiré Sidibé, Olivier Morel, Fabrice Meriaudeau, Yifei Zhang, Désiré Sidibé, Olivier Morel, Fabrice Meriaudeau, Yifei Zhang, Olivier Morel,

and Fabrice Meriaudeau. Incorporating depth information into few-shot semantic segmentation. In *25th International Conference on Pattern Recognition (ICPR 2020)*, Milan, Italy, 2021.

[285] Zhipeng Zhang and Houwen Peng. Deeper and wider siamese networks for real-time visual tracking. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:4586–4595, 2019. ISSN 10636919. doi: 10.1109/CVPR.2019.00472.

[286] Zilin Zhang. Deep-Learning-Based Early Detection of Diabetic Retinopathy on Fundus Photography Using EfficientNet. *ACM International Conference Proceeding Series*, pages 70–74, 2020. doi: 10.1145/3390557.3394303.

[287] An Zhao, Mingyu Ding, Zhiwu Lu, Tao Xiang, Yulei Niu, Jiechao Guan, Ji Rong Wen, and Ping Luo. Domain-adaptive few-shot learning. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1390–1399, Waikoloa, HI, USA, 2021. IEEE. doi: 10.1109/WACV48630.2021.00143.

[288] Bingchen Zhao and Xin Wen. Distilling Visual Priors from Self-Supervised Learning. In *European Conference on Computer Vision (ECCV 2020)*, volume 12536 LNCS, pages 422–429, Glasgow, Scotland, 2020. Springer International Publishing. ISBN 9783030660956. doi: 10.1007/978-3-030-66096-3-29. URL `http://dx.doi.org/10.1007/978-3-030-66096-3-29`.

[289] Hengshuang Zhao, Jiaya Jia, and Vladlen Koltun. Exploring Self-attention for Image Recognition. In *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 10076–10085, 2020.

[290] Yinan Zhao, Brian Price, Scott Cohen, and Danna Gurari. Objectness-Aware One-Shot Semantic Segmentation. *arXiv preprint arXiv: 2004.O2945*, 2020. ISSN 23318422. URL `http://arxiv.org/abs/2004.02945`.

[291] Fengwei Zhou, Bin Wu, and Zhenguo Li. Deep meta-learning: Learning to learn in the concept space. *arXiv preprint arXiv:1802.03596*, 2018. ISSN 23318422. URL `http://arxiv.org/abs/1802.03596`.

[292] Linjun Zhou, Peng Cui, Shiqiang Yang, Wenwu Zhu, and Qi Tian. Learning to learn image classifiers with visual analogy. In *Computer Vision and Pattern Recognition 2019*, pages 11497–11506, Long Beach, CA, USA, 2017.

[293] Chenchen Zhu, Fangyi Chen, Uzair Ahmed, Zhiqiang Shen, and Marios Savvides. Semantic Relation Reasoning for Shot-Stable Few-Shot Object Detection. In *Conference on Computer Vision and Pattern Recognition (CVPR 2021)*, 2021.

[294] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *International Conference on Learning Representations, ICLR 2021*, pages 1–16, 2021.

[295] Yousong Zhu, Chaoyang Zhao, Chenxia Han, Jinqiao Wang, Hanqing Lu, Chaoyang Zhao, Chenxia Han, Jinqiao Wang, and Hanqing Lu. Mask guided knowledge distillation for single shot detector. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1732–1737, 2019.