Universität des Saarlandes
Max-Planck-Institut für Informatik

$(SP)^2$Net for Generalized Zero-Label Semantic Segmentation

Masterarbeit im Fach Informatik
Master's Thesis in Computer Science
von / by

Anurag Das

angefertigt unter der Leitung von / supervised by
Prof. Dr. Bernt Schiele

betreut von / advised by
Prof. Dr. Bernt Schiele
Dr. Yongqin Xian
Dr. Yang He

begutachtet von / reviewers
Prof. Dr. Bernt Schiele
Dr. Yongqin Xian

Saarbrücken, June, 2021

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Statement in Lieu of an Oath

I hereby confirm under oath that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides Statt, dass die vorliegende Arbeit mit der elektronischen Version übereinstimmt.

Statement in Lieu of an Oath

I hereby confirm the congruence of the contents of the printed data and the electronic version of the thesis.

Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

_____ _____

(Datum / Date) (Unterschrift / Signature)

Acknowledgments

First and foremost, I would like to thank Prof. Bernt Schiele, for giving me the opportunity to work in this exciting project. His immense knowledge and experience helped in shaping, as well as completion of this project. I would like to express my sincere gratitude to my advisors Yongqin Xian and Yang He, for their assistance at every stage of the project. Thank you for the unwavering support and believe in me. I could not have imagined having a better advisor. I would also like to extend my sincere thanks to Prof Zeynep Akata, for her insightful comments and suggestions throughout the project.

I would like to thank the Graduate School for Computer Science for giving me the opportunity to study at Saarland University. Specifically I would like to thank Michelle and Susanne for their help and motivation throughout the GradSchool program.

I am deeply grateful to all of my friends, who have helped me grow as an individual. I am also thankful to them for making my stay at Saarbrucken a memorable one. Lastly, I am thankful to my family, who have constantly motivated and supported me throughout my studies.

$(SP)^2$Net for Generalised Zero-Label Semantic Segmentation

by

Anurag Das

Submitted to the Department of Computer Science
on June 9, 2021, in partial fulfillment of the
requirements for the degree of
Master of Science in Computer Science

Abstract

Generalized zero-label semantic segmentation aims to make pixel-level predictions for both seen and unseen classes in an image. Prior works approach this task by leveraging semantic word embeddings to learn a semantic projection layer or generate features of unseen classes. However, those methods rely on standard segmentation networks trained with large quantities of annotated data and may not generalize well to unseen classes. To address this issue, we propose to leverage a class-agnostic segmentation prior provided by superpixels and introduce a superpixel pooling (SP-pooling) module as an intermediate layer of a segmentation network. Also, while prior works ignore the pixels of unseen classes that appear in training images, we propose to minimize the log probability of seen classes alleviating biased predictions in those ignore regions. We extensively show that our $(SP)^2$Net significantly outperforms the state-of-the-art on different data splits of PASCAL VOC 2012 and PASCAL-Context benchmarks.

Thesis Supervisor: Bernt Schiele
Title: Professor

Contents

Chapter 1

Introduction

1.1 Motivation

Semantic Segmentation is one of the most fundamental computer vision task, where we try to find what is in the image and where it is located with pixel level accuracy. More specifically we classify each individual pixel of the image as given class. Since we are making predictions at the pixel level, we also call it as dense prediction task. It has a huge set of applications in real life including autonomous driving, medical image analysis and diagnosis, satellite Imagery etc. Deep learning based methods have been successful in performing semantic segmentation with quite good accuracy. But these methods require to train a model with huge set of parameters and lot of annotated training data. Also, for real world setting it is near impossible to get annotated data for all samples. We discuss these problems below :

Semantic Segmentation requires dense annotation.
Semantic Segmentation requires pixel level annotation for training data, where annotation effort is very expensive. There has been several attempts to reduce this annotation effort. One such method is to use weak supervision signals like bounding box, image level label etc, instead of pixel level annotation to semantically segment the image. Still these methods cannot be used to segment novel classes which is never seen in the training data.

Long-tail data distribution in real world scenario.
In real world setting, data distribution is long tailed. We see common classes more frequently compared to rare classes. The data distribution follows an exponential curve as shown in fig 1-1. For common classes for which sufficient samples are available we can use supervised learning methods, while for classes for which only few samples are present few-shot learning based approaches are used. For classes with no samples, zero shot learning based methods are used.

These problems were tackled first by weakly supervised learning approaches that utilize weaker forms of annotations, e.g., bounding boxes [25], key points [5], image-

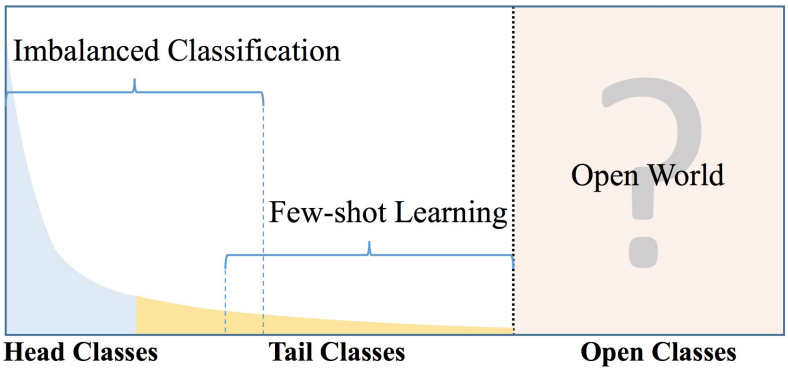


Figure 1-1: Data distribution in real world setting. For the tail classes we have very few samples, while there are classes with no samples. Image from [37]

level labels [28], and scribble-level annotations [32]. Recently, few-shot semantic segmentation methods have taken a different route by segmenting novel classes with only a few labeled training examples. However, those methods are not capable of making pixel-wise "zero-shot" predictions for the classes without a single label, which is an important real-world scenario. Therefore, successful methods in this task have the potential to significantly reduce the labeling efforts.

In the "zero-label" [61] setting, semantic segmentation models may have access to the pixels from novel classes but they do not have access to their pixel labels. In the generalized zero-label semantic segmentation (GZLSS) [61, 6, 31, 18], the goal is to make pixel-level predictions for both seen classes with abundant labels and novel classes without any label. Please note that the Generalised Zero label semantic segmentation is a challenging problem because the model trained on seen classes are highly confident on seen classes and thus performs poorly on the unseen classes. Also, this problem is realistic as during evaluation we need to segment all the present objects in an instance. In this thesis we target to solve the problem of Generalised Zero-label semantic segmentation problem.

Prior methods mainly focus on learning feature generators [6, 31, 18] or a semantic projection layer [61]. Moreover, those works rely on standard segmentation networks i.e., DeepLab-v3+ [9] trained with large quantities of annotated data. The issues faced by the prior methods are :

- **Generalisation on unseen classes.** The prior methods [6, 31, 18, 61] fail to generalise well on the unseen classes without any training samples. These methods rely on generated features from the segmentation backbone and do not work on the improvement of these generated features.

- **Biased prediction towards Seen classes.** GZLSS suffers from the severe data imbalance issue. With no training sample from the unseen class during training, the model trained is highly biased towards the unseen classes. This results in poor performance on the unseen classes. Prior method [61] solved this issue with calibration factor hyperparameter, that reduces the confidence

Figure 1-2: Top row : Zero Label Semantic Segmentation (ZLSS) task. Bottom row : Generalised Zero Label Semantic Segmentation (GZLSS) task. During training seen classes pixels are only available. During evaluation, for ZLSS task model needs to predict among novel classes only, while for the GZLSS task, model needs to predict from the complete label set, ie seen and novel classes both. Image from [10]

of seen classes during evaluation. Such method is not optimal as we cannot have a global calibration factor for all images in the evaluation.

Our main focus is to enable the segmentation networks to achieve better generalization for unseen classes along with removing this seen class baiseness during training.

1.2 Contribution

In this work, we aim to solve the problem of Generalised Zero-label Semantic Segmentation ie making pixel level predictions for both seen classes with abundant labels and novel classes without any labels. We try to achieve better generalisation on the unseen classes. At a high-level, we would like to explore superpixels [4, 41] i.e., groups of pixels that share similar visual characteristics, to learn more generic image features. While superpixels are intuitively beneficial for segmentation tasks due to their precise boundaries and context information, how and where to incorporate them into a convolutional neural network is not obvious.

We believe that aggregating features from the superpixel regions provides a generic class-agnostic segmentation prior for segmentation networks such as DeepLab-v3+ [9] and PSPNet [66]. To this end, we propose a superpixel pooling module as an intermediate layer of segmentation networks. The resulting architecture lends itself better for generalization to seen as well as unseen classes.

Furthermore, GZLSS suffers from the severe data imbalance issue, biasing the predictions towards seen classes. Therefore, we devise a simple solution to resolve this issue. Our main assumption is that the ignore regions of training images do not contain pixels from seen classes. Note that this assumption holds true according to the

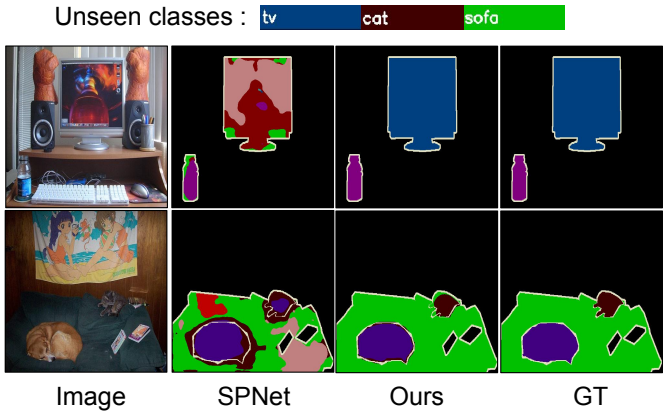


Figure 1-3: In this example, our model can predict unseen classes (tv in row 1, and sofa + cat in row 2) correctly compared to the baseline SPNet model. We integrate a novel superpixel pooling module in the segmentation network along with a bias reduction loss resulting in better generalization on the unseen classes as seen from the example.

definition of GZLSS [61] where the ignore regions include only pixels from novel classes and background. Based on this, we propose a bias reduction loss that minimizes the log-likelihood of seen class predictions in the ignore regions. The insight is to treat the unlabeled unseen classes as negatives for the seen classes and thus reduce their confidence in the pixels that definitely do not belong to them. Compared to the previous balancing strategies [6, 31], our bias reduction loss is highly efficient and allows us to train the network end-to-end in a single-stage.

Our $(SP)^2$Net augments the semantic projection network (SPNet) [61] with the proposed superpixel pooling (SP) module and bias reduction loss. On PASCAL VOC 2012, our $(SP)^2$Net improves the averaged harmonic mean mIoU (on 5 splits) of the previous state of the art by 9.8%, while on the challenging PASCAL-Context dataset, we achieve a remarkable improvement of 9.4%. We further provide an extensive model analysis and qualitative results to give insights and show the effectiveness of our approach.

Our main contribution can be summarised as :

- We achieve better generalisation on unseen classes with the help of superpixels, that provides class-agnostic segmentation prior for the segmentation network.
- We solve the problem of seen class baiseness of the trained model with the help of proposed bias-reduction loss.
- We outperform the existing state of the art on GZLSS on PASCAL VOC 2012 and PASCAL-Context datasets by significant margin.

1.3 Outline of the Thesis

The rest of the thesis is organised as follows : Chapter 2 covers the necessary background required for this thesis, Chapter 3 contains overview of the related work in

this domain, Chapter 4 explains our proposed method $(SP)^2$Net in detail, Chapter 5 provides with different experiment results and chapter 6 discusses about the summary and possible future work in this direction.

Chapter 2

Background

In this chapter we discuss the necessary background for this thesis. We begin with discussing semantic segmentation. We discuss three major CNN based semantic segmentation models, namely Fully Convolutional Network (FCN), DeepLabV3+ and PSPNet. We discuss FCN as it forms the base of all CNN based segmentation models and it is one of initial CNN based segmentation models. We then discuss DeeplabV3+ and PSPNet models as we use them in our work. We then discuss the Generalised Zero-label semantic segmentation problem, followed by Semantic Projection Network, on which our work is based.

2.1 Semantic Segmentation

Semantic segmentation is one of the most fundamental computer vision task, where pixel level labels are predicted. Since, the predictions are at the pixel level, it is also referred as dense prediction task. It has a huge array of applications ranging from robotics, autonomous driving, medical image understanding to satellite image understanding and weather predictions. Majority of the semantic segmentation models have a pretrained backbone model (for eg - Imagenet pretrained Resnet model) which provides with the feature representation at smaller scale. This comprises of the encoder part of the segmentation module. The difference in several approaches comes from how the decoder part is implemented. In this background study, we discuss three important architectures namely FCN (Fully Convolution Network) [38], DeeplabV3+ [9] and PSPNet [66] models. FCN model is the first deep learning based semantic segmentation model, which forms the base of majority of the existing semantic segmentation models. We also discuss DeepLabV3+ based model as it forms the segmentation model used in our proposal. We also discuss PSPNet as we do a comparison of performance of superpixel pooling for both DeepLabV3+ and PSPNet, showing superpixel pooling works irrespective of segmentation backbone model.

Fully Convolution Networks
As the name suggests, it is a fully convolutional network that performs pixel wise predictions. To make pixel level predictions, the last fully connected classification

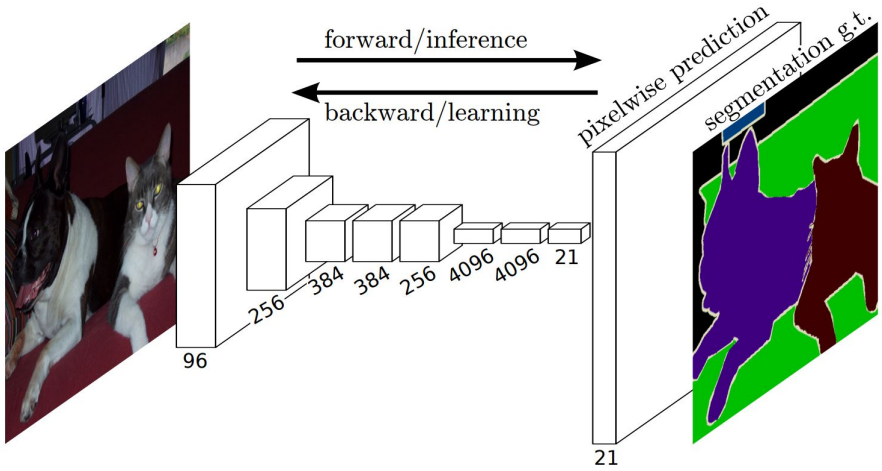


Figure 2-1: FCN based semantic segmentation. The last fully connected classification layer is replaced with convolution layer for pixel level predictions. Image from [38]

layers of the network are replaced with convolution layers (see fig 2-1). The major issue that arises is obtaining dense predictions. For dense predictions, this network uses upsampling methods such as deconvolution. The final implementation uses a stack of deconvolution layers with skip-connections from the corresponding encoder layers in order to preserve the localisation of objects and edges.

DeepLabv3+ model

FCN based semantic segmentation model suffered from fuzzy and improper object shape and boundaries due to excessive downsampling. To overcome this issue, expensive post processing steps such as CRF [26] were applied. To provide better dense representation with better object shape and boundaries, deepLab series of segmentation models were proposed. We will discuss the best performing DeepLab series model (DeeplabV3+) in this thesis as we are using it as our segmentation backbone. The Deeplab series introduced the atrous convolution which is used in almost every other semantic segmentation model. The atrous convolution is basically upsampling convolution with holes which preserves the dense representation. This improves the model performance. Further CRF based postprocessing is applied to improve upon the predictions at the boundaries. To improve upon the deepLab model, deeplabV2 was proposed that contained the Atrous Spatial Pyramid pooling (ASPP) module. Now instead of performing one Atrous Convolution, several atrous convolutions at different scales were performed on the representation obtained from the segmentation backbone. This helped in capturing context information at different scale which boosted the performance. DeeplabV3+ further improves on the DeepLabV2 by including Global Average pooling [68] in the ASPP module along with improved decoder structure. Also, it includes a connection from the intermediate encoder output to intermediate decoder output as shown in , for better object localisation. DeeplabV3+ is the best performing model among existing deeplab models.

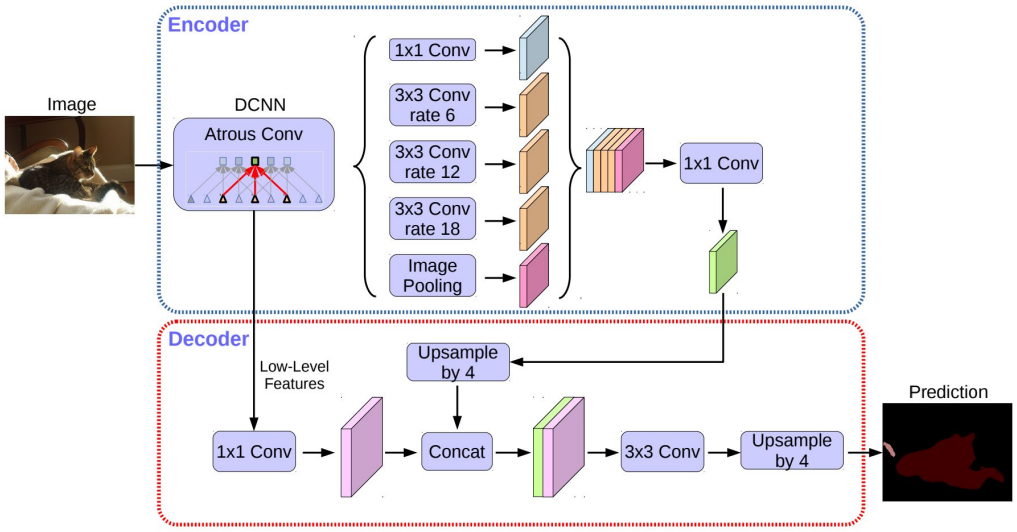


Figure 2-2: DeepLabV3+ segmentation network. The feature representations from the backbone model goes to Atrous Spatial Pyramid Pooling module which provides the contextual information at various scales. Further the decoder network helps in improving the segmentation result for the object boundaries. Image from [9]

PSPNet

PSPNet is another semantic segmentation model with significant improvement over the FCN based model. It aims to make use of the contextual relations between objects in the scene for better semantic segmentation predictions. To do so, it proposes Pyramid Pooling module. The output of the backbone segmentation model goes to the pyramid pooling module which does average pooling at different scales (see fig : 2-3) on different subregions of the image. Further it also applies global average pooling for getting global contextual prior which has been seen useful in image level classification tasks. The pooled features from different parallel layers are concatenated together, which forms the final representations. This representation goes to the decoder part which has dilated convolution layers for dense representations.

2.2 (Generalised)Zero-label semantic segmentation

For the zero-label semantic segmentation task we are required to predict pixel level labels that belong to only unseen classes. While for the generalised version, the labels can be from both seen and unseen classes (ref). This is a challenging task and in this thesis we focus on solving the generalised version task. It is important to mention that we call this task as zero-"label" and not zero-"shot" because the unseen class objects are still present during the training, but the labels for such classes are ignored. Thus,

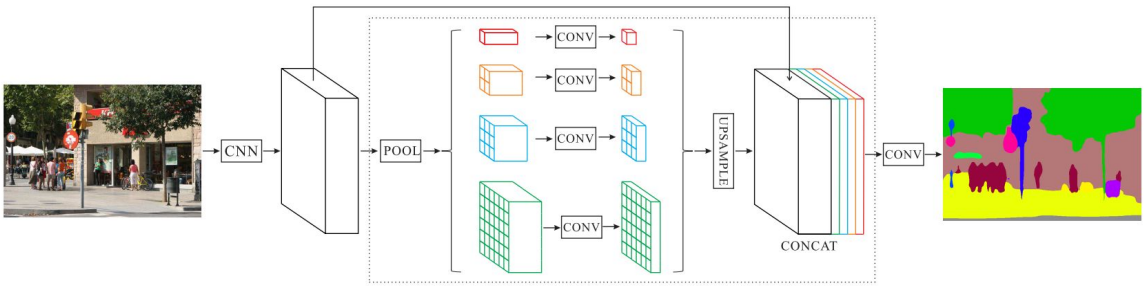


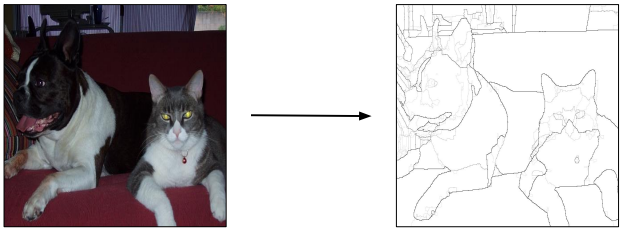
Figure 2-3: PSPNet Network Architecture. The feature representation from the backbone model is fed to the pyramid pooling module, which performs subregion average pooling along with global average pooling. This provides with the necessary contextual information at different scales. The pooled representations are concatenated which further goes to the decoder part for dense representations. [66]

we do not have apply loss function for such pixels (pixels from unseen classes) during training. We use transfer of information from the seen classes to unseen classes using the label embedding space for solving this task.

Word embeddings Word embeddings are representation of words in a fixed dimension space (generally lower dimension space). Assuming the vocabulary to be $\mathcal{V}$, we have word embedding for each word $w \in V$, where $w \in \mathbb{R}^d$, such that $d << |\mathcal{V}|$. This helps in learning the latent representations of the word such that similar words are near in the embedding space, and different words are far apart. We use word embedding space to transfer knowledge from seen to unseen classes for our GZLSS task. The class labels (both seen and unseen) are represented in a shared embedding space. Since, this embedding space contains side information about the unseen class labels, we utilise it for our GZLSS task. We use concatenation of two different type of word embeddings namely word2vec and fasttext, as it shows best result for our task (see [61]). Word2Vec and fasttext both uses context information of a word to train a model to get its representation. For word2vec, we have two ways to obtain the word embedding. One is the Continuous Bag of Word (CBOW) model where the model predicts the current word from its context [for a fixed window], and other is skip-gram model where from a given word, the model predicts its context. Fasttext is inspired from Word2Vec where each word is treated as combination of n-grams, and the corresponding vector is sum of the vector of these n-grams. This is specifically helpful for getting the embedding of "out of vocabulary" words, as it can be the sum of the constituent n-grams vectors.

2.3 Superpixels

Superpixels [2, 4, 41, 55] are a group of pixels sharing some common features eg pixel intensity. This grouping helps in reducing the computation overhead for some applications where computation on large number of pixels is expensive. Also, superpixels



Hierarchical segmented structure output of COB

image k = 0.5 k = 0.2 k = 0.1

Figure 2-4: Top row : Output of COB method, we see hierarchical segmented structures. Bottom row : Superpixels of different level with threshold parameter (k)

are far more information-rich compared to individual pixels. They have been useful in many vision tasks including object detection [52, 63, 13] , saliency [47, 20], , semantic segmentation [15], depth estimation [35, 34] etc. Superpixels can be obtained by different methods such as Watershed based (Water Pixels [39, 40], Comapact Watersheds [45]), Graph based (Normalised cut [49], Random Walk [17, 16], MCG-Multiscale Combinatorial Grouping [4]), CNN based (COB - Convolutional Oriented Boundaries [41]), clustering based (SLIC [1, 2], DASP [59]) etc. For our work we use superpixels for GZLSS task. Superpixels can provide additional shape and contextual information as prior which can improve performance of the model. We use COB based superpixels as it shows best performance [41] among other superpixels.

Retrieving different level of superpixels. The output of COB based methods is a hierarchical segmented structure as shown in 2-4. This is obtained by fusing different ultrametric contour maps (UCM), which are obtained by a single pass through the trained COB model (for more details refer [41]). The edge strength in the hierarchical segmented structure can be thresholded to give superpixels of different scale 2-4. We can see that for higher value of threshold parameter(k), we obtain lesser superpixels and the superpixels are bigger in size on average, while for lower k we obtain more superpixels with smaller sizes on average.

2.4 Semantic Projection Network

Semantic Projection Network (SPNet) is one of the initial works in the domain of Zero-/Few-shot Semantic Segmentation. It uses knowledge transfer from the text domain (label embedding space) to visual domain (pixel level representation). It projects each pixel representation obtained from the segmentation head to the label embedding space. This projection gives the class probabilities on which classification based loss function can be applied. We discuss SPNet in detail below :

Problem formulation.
We denote the set of seen classes as $\mathcal{S}$, a disjoint set of novel classes as $\mathcal{U}$ and the union of them as $\mathcal{Y} = \mathcal{S} \cup \mathcal{U}$. Let $\mathcal{T} = \{(x, y)|x \in \mathcal{X}, y_i \in \{I, \mathcal{S}\}\}$ be the training set where x is an image of spatial size $H \times W$ in the RGB image space $\mathcal{X}$, y is its label mask with the same size, and y_i is the class label at pixel x_i belonging to either one of the seen classes $\mathcal{S}$ or ignore region labeled as I. Moreover, each class label is represented by the word embedding (e.g., word2vec [42]) associated with its class name. We denote the word embedding matrices of seen and novel classes with $A^s \in \mathbb{R}^{D\times|\mathcal{S}|}$ and $A^u \in \mathbb{R}^{D\times|\mathcal{U}|}$ where D is the dimension of the word embedding. Given $\mathcal{T}$, A^s and A^u, the goal of generalized zero-label semantic segmentation (GZLSS) is to learn a model that is capable to make pixel-wise predictions among both seen and novel classes.

Semantic projection.
We follow SPNet [61] to segment novel classes via mapping pixel features into a semantic embedding space. Specifically, SPNet consists of a visual-semantic embedding module and a semantic projection layer. The former (denoted as ϕ) is based on a standard segmentation network (e.g., DeepLab-v3+ [9]), encoding each pixel x_i as a D-dimensional feature embedding $\phi(x)_i$ in the semantic embedding space. The latter computes the compatibility scores between the pixel and word embeddings followed by applying softmax that maps scores into a probability distribution,

$$P_c(x_i) = \frac{\exp s_c(x_i)}{\sum_{c'\in\mathcal{Y}} \exp s_{c'}(x_i)} \tag{2.1}$$

where $s_c(x_i) = \phi(x)_i^T a_c$ and a_c denotes the word embedding of class c. The scoring function is capable to compute the compatibility score of a given pixel to any class using its word embedding, thus enabling zero-shot prediction.

For a particular labeled training pixel (x_i, y_i) from seen classes $\mathcal{S}$, the following cross-entropy loss is optimized,

$$\mathcal{L}_C(x_i, y_i) = -\log P_{y_i}(x_i) \tag{2.2}$$

Note that the image x might include pixels from unseen classes, but those pixels are not labeled (i.e., $y_i = 0$) and their losses are ignore for ZLSS. The network can be trained in an end-to-end manner by optimizing the above loss on the whole training

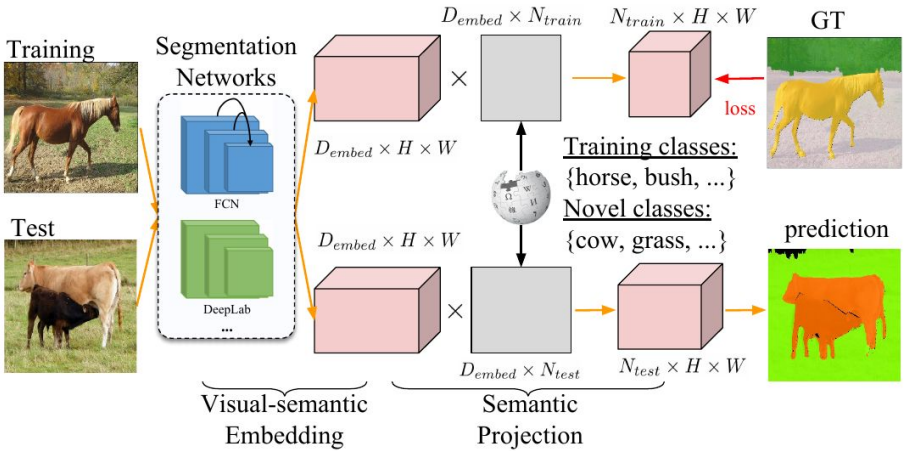


Figure 2-5: Illustration of SPNet [61] Model. It consists of two parts - Visual-semantic Embedding and Semantic Projection. We obtain pixel representation from the visual-semantic embedding module which further goes to Semantic Projection module for obtaining the class probabilities.

set $\mathcal{T}$ of seen classes.

Once, the model is trained we can inference the pixel level predictions as :

$$f_{zlss}(x_i) = \arg\max_{u \in \mathcal{U}} p(y_i = u | x_i; W^u) \tag{2.3}$$

where W^u is the label embedding of the given unseen class u.

Data imbalance issue for GZLSS

For Generalised setting, both seen and unseen classes are required to be predicted. This is a challenging task as the model trained without any samples of unseen classes, is highly biased towards predicting the seen classes and gives very low confidence score to unseen classes. To solve this issue, SPNet proposed to use a global calibration factor λ. During evaluation, the seen class confidence are reduced with this calibration factor, thus improving unseen class performance. Thus for GZLSS, the equation in 2.4 becomes :

$$f_{gzlss}(x_i) = \arg\max_{a \in \mathcal{S} \cup \mathcal{U}} p(y_i = a | x_i; W^a) - \gamma \mathbb{I}(a \in \mathcal{S}) \tag{2.4}$$

where W^a is the label embedding of the given seen/unseen class a, and $\mathbb{I}$ is the identity function, which activates only when given embedding is from seen classes. Please note that having a global calibration factor is not optimal, as class confidence is at pixel level. We propose bias reduction loss in our work which reduces this seen class biasness during training itself, thus removing the requirement of suboptimal calibration factor.

Chapter 3

Related Work

In this chapter we discuss the related work in this domain. More specifically we discuss approaches to solve semantic segmentation problem with limited labelled data. We also discuss about superpixels and it's use in solving semantic segmentation in prior works.

3.1 Zero-shot learning

Zero-shot learning aims to predict the novel classes that are not observed, by using semantic embeddings e.g., attributes [29], and word2vec [43], which encode the similarities between seen and unseen classes. Early works tackle this task by learning attribute classifiers [29, 23] or learning a compatibility function [12, 3, 60, 64, 50] between image and semantic embeddings. For the attribute classifier methods, the approach is generally two-staged, where in the attributes are predicted in first stage, and then class label is inferred as the class that has maximum similar attributes in the second stage. [30] suggests two attribute based methods specifically Direct Attribute Prediction (DAP) and Indirect Attribute Prediction (IAP). DAP based method uses a decoupling intermediate attribute layer between images and layer of labels. During training the parameters for attribute layer are learnt, while at test time the class labels are estimated by MAP. On the other hand for IAP, the attribute layer is between layer of labels (seen and unseen). During training, a classifier is learnt for each seen class, while at test time, predictions for seen classes induces labelling of attributes, which is used to infer unseen classes. CONSE [46] is another attribute based method, where the image features are learnt from the seen classes and then they are projected on label embedding space (word2Vec).

On the other hand, generalized zero-shot learning (GZSL) [7] requires the model to predict both seen and unseen classes, which is more challenging due to the extreme data imbalance issue. Since no training samples of the unseen classes are present, the learnt model is highly confident on the seen classes and performs poorly on the unseen classes. Some notable works for GZSL include generating features of novel classes [62, 27, 51, 69], and using unlabeled test data from novel classes [54] i.e., transductive ZSL. The approach where features of novel classes are generated are generally two-staged.

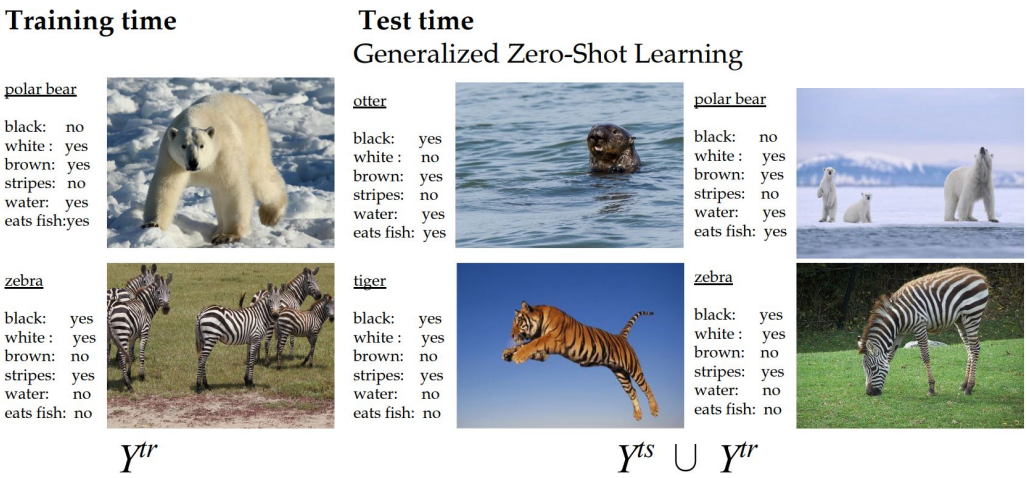


Figure 3-1: Attribute based Zero shot learning (ZSL) and Generalised Zero shot learning (GZSL). For ZSL, during training attributes and labels from seen classes are available, while during test time, only unseen class labelss are present for evaluation. Further for GZSL, training is similar to ZSL, ie only seen class labels and attributes are available, but at test time, both seen and unseen classes are present.

The features of novel classes are generated in first stage, while an off-the-shelf classifier is used to train on both seen and generated novel samples. Since, we have sufficient generated samples of novel classes, the model bias towards seen classes are removed. [27] uses conditional variational autoencoder to generate the features of the unseen labels, where the latent codes obtained by the encoder are embedded with attribute vector of the seen class during training. During evaluation, the trained model can be used to generate the novel training samples given the attributes of the novel classes. [62] uses attribute conditioned feature generating adversarial network to generate the features of the novel classes. [51] further improves on the feature generating approach, by learning a shared latent embedding space for image features and class embedding via VAE, and ensuring cross aligment of both the latent features along with distribution alignment of the latent distribution of both the modalities. [54] uses transductive ZSL to overcome the seen class biaseness issue, by assuming unlabeled novel class images are present during training. With this added information, the model makes sure, that the novel class features are away from the seen class features in the latent embedding space, during training and thus improves on the bias issue.

In contrast, we are interested in the GZLSS problem which requires making pixel-wise predictions. We focus on addressing the challenges that are specific to the semantic segmentation problem i.e., leveraging superpixels for better context modeling and alleviating imbalanced issues using ignore regions.

3.2 (Generalised) Zero-label semantic segmentation

Data annotation for segmentation is very expensive and imprecise enough for the large-scale dataset creation with numerous categories. Inspired by the success of zero-shot image classification, zero-label semantic segmentation has been proposed recently and became increasingly popular, which is able to segment a novel classes without having their annotated masks during training. ZLSS aims to segment novel classes without having their annotated masks during training. We focus on generalized zero-label semantic segmentation (GZLSS) where the model is required to segment both seen and novel classes. Prior works [61, 31, 6] tackle this task by adapting technics from zero-shot image classification into segmentation networks e.g., learning semantic projection layer [61] and generating pixel-wise features [31, 6] of novel classes, ignoring the challenges that are specific to semantic segmentation. SPNet [61] proposed to incorporate word embedding e.g., word2vec [42], into fully convolutional architectures [8] by learning a semantic projection layer. This semantic projection layer projects the visual features onto the label embedding space. During evaluation, by simply replacing embeddings for new classes, SPNet is enabled to segment unseen objects. SPNet suffers from seen class biasness issue, as the model is trained on seen class label samples only. Bucher et al. [6] developed a feature generator that synthesizes pixel-wise features for unseen classes and train a segmentation model jointly for this task. Both of these tasks are performed in two separate stages. They further try to improve on the performance with self training, by using the pseudo labels produced by the initial network for unseen classes. Hu et al. [22] suggests learning from representative samples from the seen classes, as learning from noisy samples can lead to performance drop. They provide a novel framework that identifies noisy samples, thus allowing network to learn only from representative samples. Recently, Li et al. [31] improved the feature generator by exploiting the structural relation-based constrain between seen and unseen categories. They try to capture the similarity among classes (seen and unseen) in the label embedding space, and enforce them in the generation of visual features, resulting in improvement of performance. Different from aforementioned work, we significantly improve the zero-label and few-label semantic segmentation by applying superpixel priors which provide pre-segmentation and allowing our network focus on recognizing each region or object. Even without feature generation which requires additional model parameters, we are able to achieve very competing or even better results compared with current state of the arts. Our proposed method improves the SPNet [61] by incorporating superpixels [41] into the network for learning dense features that generalize better to unseen classes. Moreover, we leverage ignore regions in training images to alleviate the imbalance issue, which is more efficient than previous feature generation methods [31, 6] and can be trained end-to-end in a single stage.

3.3 Superpixel and semantic segmentation

Superpixel [2, 4, 41, 55] and semantic segmentation [38, 8, 9, 21, 66] have a long history in computer vision, which provides a pre-segmentation and understanding of images. The convolutional encoder-decoder architectures represent images into structural feature maps and predict the class labels in the end. Generally, it requires expensive dense annotations to train the structural output models. To address this issue, superpixel has been combined with modern deep neural networks to boost semantic segmentation models trained on a variety of supervisions [14, 21, 33, 28, 25]. In particular, it shows surprisingly promising results in a weakly supervised learning setup, owing to its clustering capability on similar pixels. This property complements the lack of training signals and provides useful boundary information for weakly supervised segmentation setup, including image-level annotation [28], box-level annotation [25] and partially labeled videos [21] etc. Kwak et al. [28] performed semantic segmentation with superpixels as side information, with just image level labels. They had two staged network, wherein weak annotations were generated in the first stage with the help of superpixels, followed by a decoupled network in second stage that performs semantic segmentation with labels predicted in first stage. Different annotations of a class from first stage helps in learning better segmentation in second stage. He et al. [21] uses multi-view unlabelled frames along with superpixel information to improve on the semantic segmentation on the target frame. Different from previous works, we aim to improve GZLSS by incorporating superpixels into segmentation networks. To the best of our knowledge, we are the first to leverage superpixels in GZLSS.

3.4 Few-shot semantic segmentation

Another avenue related to our work is few-shot semantic segmentation, where it is based on meta-learning approaches and the model aims to segment objects from the same category to a support set. The models [65, 58, 48, 53, 57, 56] usually learn a similarity/distance function and search regions which match the supportive images well. Different from the above work, we will not have the labels for unseen classes; instead we only have the images of unseen classes and thus call our task zero-label segmentation. Specifically, we follow the established terminology in [61] without using support sets and perform semantic segmentation in inference like fully supervised models.

Chapter 4

Superpixel-Pooled Semantic Projection Network (SP^2Net)

We propose to incorporate a class-agnostic segmentation prior to the network to achieve better generalization on novel classes and leverage the ignore regions to address the biased prediction issue. Figure 4-1 shows an overview of our approach $(SP)^2$Net, consisting of three main components: (1) a novel superpixel pooling module to capture class-agnostic segmentation priors, (2) a semantic projection layer [61] for segmenting novel classes (introduced in Section 2.4), (3) a bias reduction loss in ignore regions. In the following, we will describe our major contributions i.e., superpixel pooling and the bias reduction loss.

4.1 Superpixel pooling module

Unlike SPNet [61] and other prior works [6, 31], we argue that relying on a standard segmentation network, may not generalize well to unseen classes. Therefore, we develop a novel superpixel pooling (SP-pooling) module which aims to facilitate feature learning for GZLSS.

Integrating superpixels.
As labeled images from unseen classes are not available in GZLSS, learning generic features or introducing prior information becomes important for achieving good generalization on unseen classes. We believe that superpixels could be particularly helpful for GZLSS because they can provide precise and generic object boundaries. We incorporate the superpixels as a pooling layer in our segmentation network. The simplest idea is to apply the superpixel pooling as a post processing step on the output probability scores of the network i.e., after the semantic projection layer (see Figure 4-1). Although this indeed yields smooth predictions, the superpixels do not benefit feature learning. An alternative is to pool the final feature map $\phi(x)$ after the segmentation head i.e., atrous spatial pyramid pooling (ASPP) module of DeepLab-v3+ [9] or PSPNet [66]. However, we found that applying SP-pooling before the ASPP yields best performance as superpixels can guide the ASPP to learn more generic class-agnostic

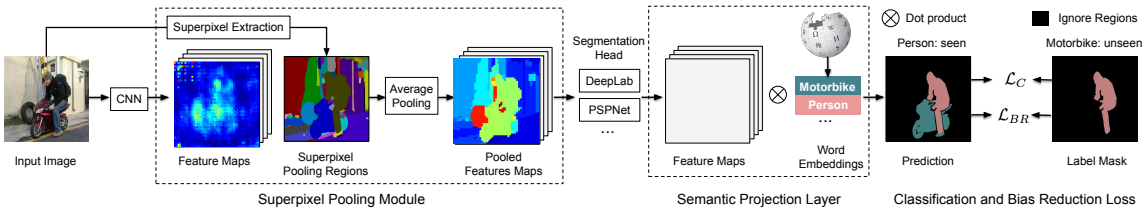


Figure 4-1: Overview of our (SP)^{2}Net for generalized zero-label semantic segmentation. We propose to capture class-agnostic segmentation priors by introducing a novel Superpixel Pooling Module, to be integrated into standard segmentation networks, followed by a semantic projection layer [61] to embed pixel features into a semantic space. Finally, we devise a bias reduction loss ($\mathcal{L}_{BR}$) to alleviate biased predictions in ignore regions that belong to unseen classes (e.g. motorcycle).

information.

Superpixel pooling.
Pooling operation plays an important role in semantic segmentation to extract global features [36] or pyramid features [67]. For efficiency, we apply simple average pooling to each superpixel region (see Figure 4-2), which is parameter-free, and not sensitive to the spatial size of the region. More specifically, given an input feature map $F_{in} \in \mathbb{R}^{K\times H\times W}$ and corresponding superpixels $\{s_n|n = 1, ..., N\}$ of the input image, we compute

$$F_{out}(k, i, j) = \frac{1}{|s_n|} \sum_{(p,q)\in s_n} F_{in}(k, p, q), \tag{4.1}$$

where (i, j) is a pixel in superpixel s_n and k is the index of feature channels. After repeating the above computation for every pixel, we obtain a pooled feature map $F_{out} \in \mathbb{R}^{K\times H\times W}$ of the same size as the input F_{in}. Note that individual input images in a mini-batch may have a different number of superpixels, but the output size of the pooling operation does not depend on that number as Equation 4.1 assigns each pixel within the same superpixel with the same feature embedding. Our superpixel pooling module not only provides boundary information, but also context information as prior for learning better representations to segment novel classes.

4.2 Superpixel Correction

We provide details on how we properly leverage the superpixel prior, which is not perfect and is able to introduce noises into network potentially. The superpixels do not always capture the useful regions for recognition and segmentation. On the one hand, it tends to provide too small regions if we set a very small scale factor K, which is lack of enough context and brings less improvement. On the other hand, a superpixel may group multiple classes into one region, which will introduce noises to the network and lead to negative effects on training. Therefore, we apply a superpixel correction on the original superpixels, which helps us to deal with imperfect superpixel

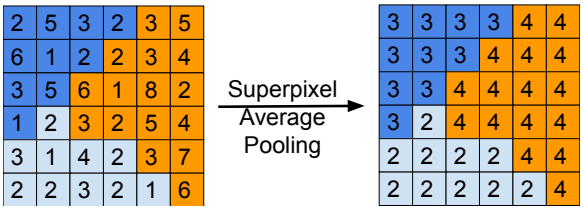


Figure 4-2: Illustration of our superpixel pooling. The color indicates individual superpixel region.

computation.

Let the 2-d array SP be a superpixel partition for a given image x with s_n being the n-th region where $s_n = \{(i,j)|SP_{ij} == n\}$. We are able to inspect the GT labels y inside s_n . We regard the regions s_n satisfying the following condition as success cases:

$$\frac{\max_{l \in \mathcal{S}}(|s_n \cap (y == l)|)}{|s_n|} > \alpha, \tag{4.2}$$

We do superpixel pooling in these success regions and use individual pixel features otherwise. As illustration in Figure 4-3, we will use the pixel features for a large superpixel by considering the ground truth labels as Eq. (4.2). It helps us to utilize wider context information as well as avoid noisy superpixel priors. In particular, we only apply this superpixel correction during training that we touch the ground truth. We will still use the superpixels without correction at the same scale factor K in inference. Even though the superpixels in inference are not perfect, we are still able to boost the baseline model significantly according to our study.

4.3 Bias reduction loss

In this section, we first explain ignore regions followed by introducing our bias reduction loss to alleviate the imbalance issue in GZLSS.

Ignore regions.
In zero-shot image classification, the training set must exclude any image from novel classes to satisfy the "zero-shot" assumption. In contrast, in semantic segmentation, where an image consists of dense labels from multiple object classes, it is not realistic to build a training set that contains no pixels from novel classes due to a large number of class co-occurrences. Therefore, a common practice in GZLSS is to follow the training set of supervised learning but ignoring the pixels from novel classes. Note that prior work [61] allows the models to process those pixels i.e., during the forward pass, but do not apply any loss on them as their labels are not accessible.

Bias reduction loss.
We argue that ignoring certain regions in the image severely biases the predictions.

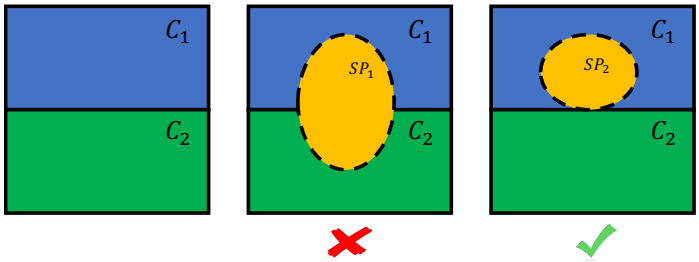


Figure 4-3: A toy illustration of superpixel correction. SP_1 contains comparable numbers of pixels from two classes C_1 and C_2. In contrast, SP_2 only contains the C_1 class, and we regard SP_2 as a success case.

Indeed, a DNN trained with only seen classes will be overconfident even on regions of novel classes at test time. Although their labels are not available, it is certain that the ignore regions do not belong to the seen classes, which can serve as strong prior information for alleviating the bias issue. Formally, for a particular unlabeled pixel x_i from the ignore regions i.e., $y_i \notin \mathcal{S}$, we propose the following bias reduction loss,

$$\mathcal{L}_{BR}(x_i) = \sum_{c \in \mathcal{S}} -\log(1 - P_c(x_i)) \tag{4.3}$$

where P_c denotes the probability of pixel x_i being predicted as class c, defined in Equation 2.1. This loss essentially treats seen classes as negative in the ignore regions and reduces the probability of those pixels being classified as any of the seen classes.
Discussion. This loss enjoys several advantages over existing balancing strategies for GZLSS. SPNet [61] adopts a post hoc calibration technique that reduces the scores of seen classes by a constant factor at the test time. However, tuning a global calibration factor that works for all pixels is extremely hard because the optimal factors for different pixels can be completely different. In contrast, we learn to alleviate the bias issue from the training data, resulting in a model that does not require any calibration at test time. Compared to feature generation approaches [6, 31], our bias reduction loss is conceptually simpler and can be optimized end-to-end. Specifically, those approaches require an additional training stage to learn the feature generator and novel class classifiers. Despite the simplicity, we outperform them [6, 31] significantly as shown experimentally.

4.4 Training and inference

As a preprocessing step, we first compute superpixels for each image with an off-the-shelf superpixel method (more details in Section. **??**). Our full model (SP)^{2}Net then minimizes the following objective:

$$\sum_{i=0}^{H \times W} \mathbb{1}[y_i \in \mathcal{S}] \mathcal{L}_C(x_i, y_i) + \mathbb{1}[y_i \notin \mathcal{S}] \lambda \mathcal{L}_{BR}(x_i) \tag{4.4}$$

where $\mathbb{1}[y_i \in \mathcal{S}]$ denotes an indicator function (= 1 if $y_i \in \mathcal{S}$ otherwise 0). $\mathcal{L}_C$ is the classification loss defined in Equation 2.2 learns the semantic projection layer on pixels of seen classes. $\mathcal{L}_{BR}$ is the bias reduction loss defined in Equation 4.3 which handles the biased prediction issue in ignore regions. λ is hyperparameter to tune, controlling the trade-off between learning semantic projection and bias reduction. Our proposed superpixel pooling layer and bias reduction loss are both differentiable, which allows us to train the model end-to-end.

Once trained, we make a prediction by searching for the class with the highest probability among both seen and unseen classes, i.e. $\arg\max_{c \in \mathcal{Y}} P_c(x_i)$.

Chapter 5

Experiments

In this chapter, we first describe our experimental setting, then we present (1) our results comparing with the state-of-the-art for the GZLSS task on 10 different data splits from two benchmark datasets, (2) model analysis on each model component and impact of hyperparameters, (3) our qualitative results comparing with SPNet.

Implementation details
We use PASCAL VOC 2012 [11] (10582 train / 1449 val images from 20 classes) and PASCAL-Context [44] (4998 train / 5105 val images from 59 classes) datasets. We adopt the same data splits used by ZS3Net [6] and CSRL [31] for a fair comparison. Specifically, we use 5 different splits for both datasets which have 2, 4, 6, 8, and 10 unseen classes in an increment manner respectively. For PASCAL VOC 2012, the unseen classes are 2-cow+motorbike, 4-airplane+sofa, 6-cat/tv, 8-train+bottle and 10-chair+potted-plant. The unseen classes are incremental since unseen classes from split 2 are contained in split 4, which are further present in split 6 and so on. Similarly for PASCAL-Context we have unseen classes as 2-cow+motorbike, 4-sofa+cat, 6-boat+fence, 8-bird+tvmonitor and 10-keyboard+airplane.

Unless otherwise stated, we follow ZS3Net [6] and CSRL [31] to use DeepLab-v3+ [9] with the ImageNet-pretrained ResNet-101 [19] as the backbone for a fair comparison. For the semantic embeddings, we use the concatenations of fasttext [24] and word2vec [42] embeddings (each with dimension 300) because of its superior performance as shown in SPNet [61]. We adopt the SGD optimizer with initial learning rate 2.5×10^{-4} and use "poly" learning rate decay [8] with power=0.9. We set momentum and weight decay rate to 0.9 and 0.0005 respectively. Unless otherwise stated, we apply our superpixel pooling module before the ASPP layer in DeepLab-v3+ as it performs better.

Superpixel extraction and correction.
We employ a pretrained convolutional oriented boundaries (COB) [41] provided by the authors to compute the superpixels because it is computationally efficient and generalizes well to unseen categories.

More specifically, we compute the boundary probability with COB followed by applying a threshold (K) to obtain the superpixels of different scales. A higher value

of K implies larger superpixels that are noisy, while a lower value means smaller superpixels. Unless otherwise stated, we use K = 0.2 for our experiments. Having a higher threshold result in learning from noisy signals, while lower threshold restricts the context information.

We observe that superpixels can be noisy and propose to fix the issue by a simple heuristic: ignoring the superpixels that do not intersect with the ground truth label mask sufficiently. We take the intersection of superpixel mask with ground truth label masks, and decide on whether to keep the superpixel region for pooling, if the superpixel's intersection with any ground truth label mask is greater than a given threshold. Usually we keep high threshold values to avoid noisy superpixels. Note that such correction is only applied during the training phase and not in inference as the ground truth is not available for test images.

Evaluation metric.
We compute mean Intersection of Union (mIoU) since it is widely used for semantic segmentation [8, 9]. We follow [61, 6, 31] to report mIoU for seen classes (S), unseen classes (U) and harmonic mean (HM) of both, which is computed as

$$HM = \frac{2 * mIoU_{seen} * mIoU_{unseen}}{mIoU_{seen} + mIoU_{unseen}} \tag{5.1}$$

The HM measures how well the model balances seen and unseen mIoU, i.e. the HM would be high if both are high. It is also a better evaluation metric for our task, compared to overall mIoU (which is mean of all classes) because it can affect the performance score when one of the scores (seen mIoU or unseen mIou) is low, which is not the case for overall mIoU.

5.1 Comparing with State-of-the-Art

Splits	Method	PASCAL VOC 2012			PASCAL-Context		
		Seen	**Unseen**	**HM**	**Seen**	**Unseen**	**HM**
U-2	ZS3Net [6]	72.0	35.4	47.5	41.6	21.6	28.4
	CSRL [31]	**73.4**	45.7	56.3	41.9	27.8	33.4
	SPNet [61]	72.0	24.2	36.3	**42.9**	4.6	8.3
	SPNet + $\mathcal{L}_{BR}$	69.1	40.4	51.0	34.4	16.2	22.1
	$(SP)^2$Net	**73.4**	**71.3**	**72.4**	38.1	**52.9**	**44.3**
U-4	ZS3Net [6]	66.4	23.2	34.4	37.2	24.9	29.8
	CSRL [31]	69.8	31.7	43.6	**39.8**	23.9	29.9
	SPNet [61]	70.3	32.1	44.1	36.0	11.1	17.0
	SPNet + $\mathcal{L}_{BR}$	61.3	**44.8**	**51.8**	38.0	23.8	29.3
	$(SP)^2$Net	**81.7**	37.9	**51.8**	34.6	**47.0**	**39.9**
U-6	ZS3Net [6]	47.3	24.2	32.0	32.1	20.7	25.2
	CSRL [31]	66.2	29.4	40.7	35.5	22.0	27.2
	SPNet [61]	70.3	26.6	38.6	36.3	11.5	17.4
	SPNet + $\mathcal{L}_{BR}$	71.5	36.5	48.4	**39.5**	16.7	23.4
	$(SP)^2$Net	**78.6**	**52.8**	**63.1**	35.6	**45.1**	**39.8**
U-8	ZS3Net [6]	29.2	22.9	25.7	20.9	16.0	18.1
	CSRL [31]	62.4	26.4	37.6	31.7	18.1	23.0
	SPNet [61]	66.2	22.7	33.9	**33.2**	12.6	18.3
	SPNet + $\mathcal{L}_{BR}$	65.6	23.1	34.2	26.3	17.1	20.7
	$(SP)^2$Net	**72.8**	**27.6**	**40.1**	33.1	**26.8**	**29.6**
U-10	ZS3Net [6]	33.9	18.9	23.6	20.8	12.7	15.8
	CSRL [31]	59.2	**21.0**	31.0	29.4	14.6	19.5
	SPNet [61]	68.8	17.9	28.4	30.3	10.0	15.0
	SPNet + $\mathcal{L}_{BR}$	**75.1**	16.3	26.8	33.5	12.8	18.6
	$(SP)^2$Net	73.2	19.7	**31.0**	**36.6**	**20.7**	**26.4**

Table 5.1: Comparing with the state-of-the-art methods on the generalised zero-label semantic segmentation task on 10 different splits (U-k: split with k unseen classes) of PASCAL VOC 2012 and PASCAL-Context datasets. We report mIoU (in %) on seen classes (S), unseen classes (U) and harmonic mean of them (HM).

We compare our $(SP)^2$Net to the following methods.

- SPNet [61] embeds pixel features into a semantic space and produces a probability distribution with softmax.
- ZS3Net [6] first trains a standard DeepLabv3+ model on the training set followed by a feature generator which synthesizes pixel-wise features of novel classes using their word embeddings. Finally, novel class classifiers are learned with the generated features to fix the imbalance issue.
- CSRL [31] augments ZS3Net by a structural feature generator that relates seen and novel classes. It is currently the state of the art in GZLSS on both our benchmarks.

We evaluate SPNet ourselves as it is not evaluated on the same benchmark, while results of ZS3Net and CSRL are directly taken from the papers. In addition, we combine SPNet and our bias reduction loss i.e., SPNet + $\mathcal{L}_{BR}$.

We report the results of generalized zero-label semantic segmentation under 5 different data splits of PASCAL VOC dataset in Table 5.1 (left). First, our $(SP)^2$Net outperforms feature generation approaches (i.e., CSRL and ZS3Net) significantly in

almost all cases in terms of harmonic mean mIoU (HM). In particular, on the Unseen-2 split, we achieve a remarkable HM of 72.4%, improving the state-of-the-art method CSRL (56.3%) by 16.1%. Compared to CSRL, our $(SP)^2$Net not only improves the unseen mIoU by a wide margin, but also boosts the seen mIoU in most of cases e.g., we obtain 78.6 seen mIoU (v.s. 66.2 of CSRL) and 52.8 unseen mIoU (v.s. 29.4% of CSRL) on Unseen-6 split. These results indicate that our $(SP)^2$Net generalizes well to segment both seen as well unseen classes by incorporating the super-pixel pooling module and bias reduction loss.

Moreover, we observe that our $(SP)^2$Net outperforms SPNet + $\mathcal{L}_{BR}$ significantly in 4 out of 5 cases on PASCAL VOC dataset in terms of HM e.g., 72.4% of ours v.s. 51.0% of SPNet + $\mathcal{L}_{BR}$ on Unseen-2 split, 63.1% of ours v.s. 48.4% of SPNet + $\mathcal{L}_{BR}$ on Unseen-6 split. These compelling results clearly show the importance of our superpixel pooling module. Indeed, by integrating the class-agnostic segmentation prior from superpixels, our $(SP)^2$Net learns dense image features that are more suitable for GZLSS. Another observation is that SPNet + $\mathcal{L}_{BR}$ improves the HM of SPNet in almost all cases and even surprisingly outperforms CSRL in some cases, confirming that our bias reduction loss $\mathcal{L}_{BR}$ is able to alleviates the strong bias towards seen classes.

In addition, on five data splits with different level of difficulties, our $(SP)^2$Net consistently outperforms other methods in terms of HM, establishing a new state-of-the-art on PASCAL VOC. These results are encouraging because our $(SP)^2$Net is substantially simpler and can be trained end-to-end in a single-stage. In contrast, CSRL and ZS3Net employ a three-stage learning algorithm where the segmentation backbone, feature generator and classifiers are learned in isolation. It is worth noting that CSRL, ZS3Net and SPNet indeed process the ignore regions but they fail to apply any loss on those pixels. However, our $(SP)^2$Net employs the bias reduction loss which makes full use of the ignore regions for balancing the model.

Finally, the GZLSS results on the PASCAL Context dataset are shown in Table 5.1 (right). Our $(SP)^2$Net again outperforms the state-of-the-art methods consistently on its five data splits by a wide margin in terms of HM. Although this dataset is more challenging than PASCAL VOC, our $(SP)^2$Net still achieves large performance gains over the closest baseline CSRL i.e., +10% on U-4 split, +12.6% on U-6 split, +6.6% on U-8 split, and +6.9% on U-10 split. These results indicate that our $(SP)^2$Net is able to handle complex scenes with diverse classes, which is partially due to our superpixel pooling module that enables better representation learning. Although other methods perform better on seen classes, they suffer from a significant performance drop on the unseen class mIoU. This supports our claim that SPNet, CSRL and ZS3Net may overfit to seen classes as they rely on the standard DeepLab-v3+ [9]. Our $(SP)^2$Net generalizes better to novel classes because it leverages the class-agnostic segmentation prior provided by superpixels.

5.2 Model analysis

Splits	Location of SP-pooling	PASCAL VOC 2012			PASCAL-Context		
		S	U	H	S	U	H
U-2	w/o SP-pooling	69.1	40.4	51.0	34.4	16.2	22.1
	output layer	73.7	52.5	61.3	35.9	18.0	24.0
	after ASPP	81.8	70.9	**76.0**	44.8	56.3	**49.9**
	before ASPP	73.4	71.3	72.4	38.1	52.9	44.3
U-4	w/o SP-pooling	61.3	44.8	51.8	38.0	23.8	29.3
	output layer	62.5	47.6	54.0	39.8	25.3	30.9
	after ASPP	76.1	45.5	**56.9**	34.4	40.0	37.0
	before ASPP	81.7	37.9	51.8	34.6	47.0	**39.9**
U-6	w/o SP-pooling	71.5	36.5	48.4	39.5	16.7	23.4
	output layer	72.4	37.8	49.7	41.4	18.2	25.2
	after ASPP	71.4	37.3	49.1	38.5	17.1	23.7
	before ASPP	78.6	52.8	**63.1**	35.6	45.1	**39.8**
U-8	w/o SP-pooling	65.6	23.1	34.2	26.3	17.1	20.7
	output layer	67.3	24.7	36.1	27.5	19.0	22.4
	after ASPP	72.9	27.1	39.5	35.4	20.2	25.7
	before ASPP	72.8	27.6	**40.1**	33.1	26.8	**29.6**
U-10	w/o SP-pooling	75.1	16.3	26.8	33.5	12.8	18.6
	output layer	76.5	16.1	26.6	34.6	14.2	20.1
	after ASPP	77.0	12.9	22.2	35.9	17.5	23.6
	before ASPP	73.2	19.7	**31.0**	36.6	20.7	**26.4**

Table 5.2: Applying superpixel pooling (SP-pooling) module at different layers of DeepLab-v3+. We report mIoU of seen classes (S), unseen classes (U) and harmonic mean of them (H).

In this section, we conduct extensive ablation experiments to show the effectiveness of our network design and impact of hyperparameters. Here, we report results for all splits of PASCAL VOC 2012 and PASCAL-Context datasets.

Location of superpixel pooling.
We provide results for superpixel pooling at different layers of the segmentation network (i.e., DeepLab-v3+ [9]) on all splits of both PASCAL VOC 2012 and PASCAL-Context datasets in Table 5.2. We tried three different locations for DeepLab-v3+ i.e., on the output probability scores of the output layer, on the output feature maps of its Atrous Spatial Pyramid Pooling (ASPP) module, and on the input feature maps of the ASPP module (output of the segmentation backbone of DeepLab-v3+). Please note that superpixel pooling at the output layer simply does pooling of the logit scores and is just a post-processing step on the predictions during evaluation. The output feature maps of ASPP consist of the concatenated feature maps obtained after parallel atrous convolutions of different scale on the features obtained form the ResNet backbone. On the other hand, in the input to ASPP are the visual features obtained by the last convolutional layer of Resnet backbone.

In general, all three variants of our superpixel pooling module improve the segmentation network compared to without using SP-pooling, confirming again the advantage

of using superpixels. In addition, we observe that applying SP-pooling in the intermediate layers (i.e., before and after ASPP) outperforms pooling the output, implying that SP-pooling improves feature learning. We see around 13 percent improvement in using superpixel pooling in PASCAL VOC 2012 dataset compared to around 14 percent improvement in PASCAL-Context dataset. Finally, we observe that applying superpixel pooling before the ASPP module performs the best in 7 out of 10 cases, confirming that superpixels provide class-agnostic shape priors for the ASPP.

We also observe that superpixel pooling at the output layer performs worse compared to superpixel pooling of the intermediate layers. This is understandable as pooling the outputs or last layer simply smooths the predictions or has limited effects on improving the feature learning. On the other hand, pooling before ASPP provides the ASPP module the class-agnostic segmentation prior to learn better features.

Splits	**Method** **SP-pooling**	**Segmentation** **Network**	**PASCAL VOC 2012** **S**	**U**	**H**	**PASCAL-Context** **S**	**U**	**H**
U-2	w/o	DeepLab-v3+	69.1	40.4	51.0	34.4	16.2	22.1
	w/	DeepLab-v3+	73.4	71.3	**72.4**	38.1	52.9	**44.3**
	w/o	PSPNet	78.8	25.2	38.2	38.5	8.5	13.9
	w/	PSPNet	79.6	27.7	**41.1**	43.4	18.9	**26.4**
U-4	w/o	DeepLab-v3+	61.3	44.8	**51.8**	38.0	23.8	29.3
	w/	DeepLab-v3+	81.7	37.9	**51.8**	34.6	47.0	**39.9**
	w/o	PSPNet	66.7	32.8	44.0	38.4	16.4	22.9
	w/	PSPNet	75.7	33.5	**46.5**	41.4	19.9	**26.9**
U-6	w/o	DeepLab-v3+	71.5	36.5	48.4	39.5	16.7	23.4
	w/	DeepLab-v3+	78.6	52.8	**63.1**	35.6	45.1	**39.8**
	w/o	PSPNet	69.7	22.3	33.8	37.1	12.3	18.5
	w/	PSPNet	70.5	25.5	**37.4**	39.1	14.8	**21.5**
U-8	w/o	DeepLab-v3+	65.6	23.1	34.2	26.3	17.1	20.7
	w/	DeepLab-v3+	72.8	27.6	**40.1**	33.1	26.8	**29.6**
	w/o	PSPNet	60.7	19.9	30.0	35.4	12.0	17.9
	w/	PSPNet	68.2	24.5	**36.0**	41.0	14.7	**21.7**
U-10	w/o	DeepLab-v3+	75.1	16.3	26.8	33.5	12.8	18.6
	w/	DeepLab-v3+	73.2	19.7	**31.0**	36.6	20.7	**26.4**
	w/o	PSPNet	50.6	16.3	24.6	34.1	10.9	16.6
	w/	PSPNet	73.8	15.1	**25.1**	40.4	12.9	**19.6**

Table 5.3: Superpixel pooling module (SP-pooling) based on two segmentation networks i.e., DeepLab-v3+ [9] and PSPNet [66]. We report full results under all splits of PASCAL VOC 2012 and PASCAL-Context datasets. w/o - without SP-pooling, w/ - with SP-pooling

Effect of segmentation networks.

The goal of this study is to determine the effect of using different segmentation networks that provide critical segmentation heads on top of the CNN backbone. We investigate two popular segmentation networks i.e., DeepLab-v3+ [9] and PSPNet [66] (both are with ResNet101 as the backbone). With same backbone, the major difference in both of these networks lies in the way treat the output features of the backbone to capture the contextual information. DeepLab-v3+ employs ASPP module which does parallel atrous pooling at different scales and concatenate the result,

while PSPNet does simple average parallel pooling at different scales and then sum up the features obtained. DeepLab-V3+ further employs a decoder network that tries to improve on the dense representations.

In Table 5.3 we show the this result under all splits of PASCAL VOC 2012 and PASCAL-Context datasets. For both DeepLab-v3+ and PSPNet, we observe that our SP-pooling significantly improves the performance (in terms of harmonic mean) without SP-pooling under all splits. These results demonstrate that our superpixel pooling is an effective architecture change to improve GZLSS performance and is not only limited to DeeplabV3+ backbone architecture. We further observe that improvement is much better in DeepLab-v3+ compared to PSPNet suggesting DeepLab-v3+ as better segmentation model for our GZLSS task.

Splits	**Method**	**PASCAL VOC 2012**			**PASCAL-Context**		
		S	**U**	**H**	**S**	**U**	**H**
U-2	SPNet	72.0	24.2	36.3	42.9	4.6	8.3
	+ SP-Pooling	78.4	26.2	39.3	40.0	7.3	12.3
	+ $\mathcal{L}_{BR}$	69.1	40.4	51.0	34.4	16.2	22.1
	+ SP-Pooling + $\mathcal{L}_{BR}$	73.4	71.3	**72.4**	38.1	52.9	**44.3**
U-4	SPNet	70.3	32.1	44.1	36.0	11.1	17.0
	+ SP-Pooling	79.6	35.4	49.0	38.2	11.9	18.2
	+ $\mathcal{L}_{BR}$	61.3	44.8	**51.8**	38.0	23.8	29.3
	+ SP-Pooling + $\mathcal{L}_{BR}$	81.7	37.9	**51.8**	34.6	47.0	**39.9**
U-6	SPNet	70.3	26.6	38.6	36.3	11.5	17.4
	+ SP-Pooling	70.0	28.2	40.2	39.4	12.7	19.2
	+ $\mathcal{L}_{BR}$	71.5	36.5	48.4	39.5	16.7	23.4
	+ SP-Pooling + $\mathcal{L}_{BR}$	78.6	52.8	**63.1**	35.6	45.1	**39.8**
U-8	SPNet	66.2	22.7	33.9	33.2	12.6	18.3
	+ SP-Pooling	61.6	23.3	33.9	33.6	14.3	20.2
	+ $\mathcal{L}_{BR}$	65.6	23.1	34.2	26.3	17.1	20.7
	+ SP-Pooling + $\mathcal{L}_{BR}$	72.8	27.6	**40.1**	33.1	26.8	**29.6**
U-10	SPNet	68.8	17.9	28.4	30.3	10.0	15.0
	+ SP-Pooling	65.3	19.1	29.6	28.8	10.7	15.6
	+ $\mathcal{L}_{BR}$	75.1	16.3	26.8	33.5	12.8	18.6
	+ SP-Pooling + $\mathcal{L}_{BR}$	73.2	19.7	**31.0**	36.6	20.7	**26.4**

Table 5.4: Ablation result for Superpixel pooling module and bias reduction loss on PASCAL VOC 2012 and PASCAL-Context datasets.

Ablations on superpixels and bias reduction loss.

We perform ablation studies with respect to our superpixel pooling and bias reduction loss and report the full results under all the splits in Table 5.4. We start from our baseline SPNet [61] and gradually add components to it. We first add superpixel pooling, we then add bias reduction loss to the baseline and then show results on our full model.

First, we observe that our superpixel module, i.e. SPNet + SP, improves the seen, unseen and harmonic mean mIoU of SPNet significantly on both datasets. We observe an improvement of 2.1 % on harmonic mIoU on PASCAL VOC 2012 and about 1.9 % improvement on harmonic mIoU on PASCAL-Context dataset. This again confirms our claims that the superpixel provide a strong class-agnostic segmentation prior and is able to facilitate the network to learn better dense image features for the seen as

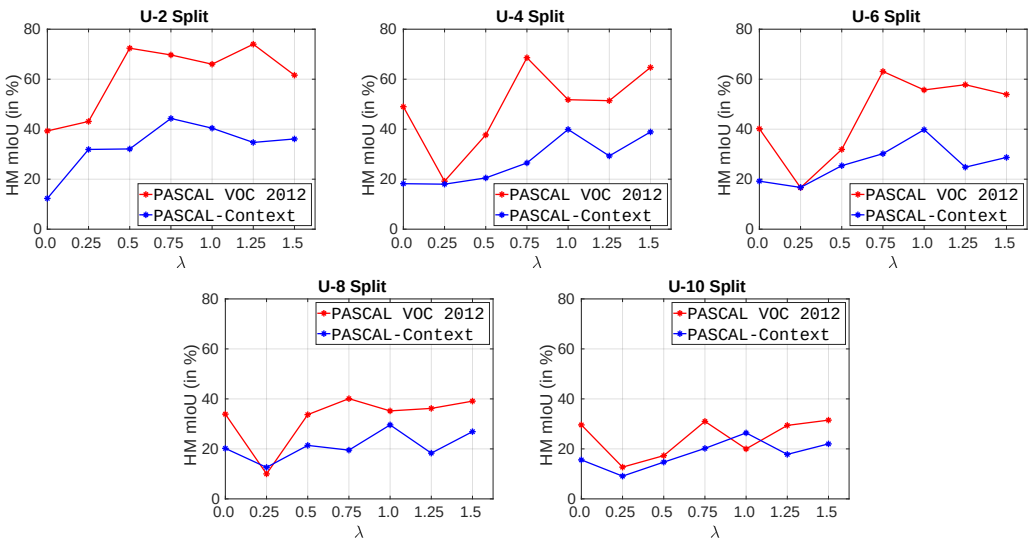


Figure 5-1: Effect of using different λ under all splits of PASCAL VOC 2012 and PASCAL-Context datasets.

well novel classes. Second, adding the bias reduction loss $\mathcal{L}_{BR}$ to SPNet immediately leads to a huge boost on the harmonic mean on both datasets, implying that the bias reduction loss is an effective way to alleviate the imbalance issue. We observe around 5.5 % harmonic mIoU improvement on average on PASCAL VOC 2012 and around 7.6 % harmonic mIoU improvement on average on PASCAL-Context dataset. Finally, putting all components together i.e., SPNet + SP + $\mathcal{L}_{BR}$, yields our full model $(\text{SP})^2$Net which outperforms all other baselines consistently (average gain of 15.5 % and 20.8 % on Harmonic mIoU on PASCAL VOC 2012 and PASCAL-context respectively wrt SPNet) on all the metrics and datasets and indicates the complements of the superpixel pooling and bias reduction loss.

Impact of hyperparameter λ.

Hyerparameter λ is associated with the bais reduction loss which tries reduce the seen class bias prediction of trained models and thus solving the data imbalance issue. In Equation 4.4, λ plays a trade-off between the classification loss $\mathcal{L}_C$ and bias reduction loss $\mathcal{L}_{BR}$. Variation in different values of λ results in different behavior of the network. We show the results of using different λ in Figure 5-1 (left) and observe the following. (1) Without the bias reduction loss ($\lambda = 0$), the harmonic mean is rather low, indicating that the model fails to handle the imbalance issue well. This model is equivalent to SPNet with superpixel pooling with no bias reduction loss. (2) using a lambda in the range of $[0.75, 1.5]$ significantly boosts the harmonic mean, confirming the effectiveness of our bias reduction loss. (3) it seems to have a trend of performance drop with a large λ because the model would fail to learn a precise semantic projection layer when putting too much weights on the bias reduction loss. The network would focus more on bias reduction loss in such case and would try to minimise confidence of seen classes in ignore regions and overlook the cross entropy

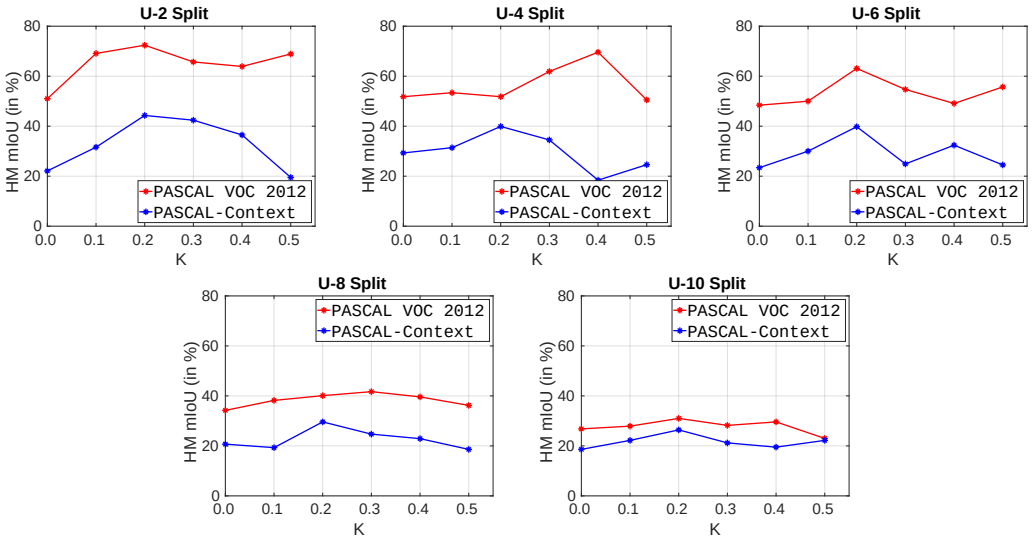


Figure 5-2: Effect of using different K (right column) under all splits of PASCAL VOC 2012 and PASCAL-Context datasets.

loss, resulting in poorer performance.

Impact of hyperparameter K.
We use the pretrained COB [41] with a threshold K to extract superpixels at different scales. This thresholding is applied at output of COB, which is the hierarchical segmented structure. The different hierarchical edges are denoted by probability scores which can be thresholded giving superpixels of different scale. A larger K indicates coarser superpixels, which provide larger context regions but may introduce more noise. For higher K, smaller superpixels join together to form a bigger superpixel, and sometimes they can have intersections with multiple classes. This introduces noise and the superpixels are not perfect. On the other hand smaller K means finer superpixels which are comparatively smaller in shape (on average) compared to the ones with higher K. Figure 5-1 (right) shows that increasing K from 0.0 to 0.2 leads to a significant performance boost on both datasets, confirming that our superpixel pooling is indeed helpful to learn generic features for unseen classes. But the performance decreases by further increasing K, which can be explained by the wrong pooling regions from noisy superpixels.

5.3 Qualitative results

We show our qualitative results in Figure 5-3 for the unseen-6 split of PASCAL VOC 2012 and PASCAL-Context datasets. We observe from the figure that unseen classes are not correctly classified by the baseline SPNet. With the introduction of bias reduction loss, the performance further improves compared to SPNet for most of the samples. Our proposed model, further improves on top of (SPNet + $\mathcal{L}_{BR}$) giving best qualitative image compared to baselines.

For PASCAL VOC 2012, we highlight the following unseen classes that our models segment much better than the SPNet: sofa (row 1, row 2), cat (row 2, row 5), aeroplane (row 2), tv (row 3), cow (row 4, row 5) and motorbike (row 3), and point out several observations: (1) By introducing bias reduction loss with SPNet (SPNet + $\mathcal{L}_{BR}$), we are able to overcome the biasness to the seen classes and then successfully recognize some unseen objects such as tv, sofa etc, although the output masks remain to improve further. In particular, in first row, we observe part of sofa being incorrectly predicted as motorbike, which is corrected by the introduction of bias reduction loss. Similar examples include sofa again in second row, tv in first row, cat fifth row and cow in fifth row, where part of the novel object is incorrectly predicted, which is further corrected by bias reduction loss. (2) Superpixel plays an important role for stronger performance w.r.t more smooth prediction over objects (e.g., the areoplane in the second row, cow in fourth row, cat in second row,) as well as learning better representations for zero-label segmentation (e.g., the cat in the second, fifth row, cow in fourth, fifth row). The smoothing over object shape is result of the average superpixel pooling, enabling better prediction across the boundaries of the objects. (3) Because of the challenges of zero-label setup, objects from seen classes are also imperfectly predicted, for example, the bottle in the first row is partially predicted as the unseen class motorbike. In contrast, our full model with superpixels segments the entire outline and outputs more favorable masks than SPNet and SPNet + $\mathcal{L}_{BR}$.

In the PASCAL-Context dataset, models also predict fine-grained classes for the background including water, wall, grass etc. From the figure, we clearly see our model not only produces more precise results for the object class but also emits smooth and accurate background segmentation. We highlight the following unseen classes which our model segments much better than SPNet : boat (row 1), cow (row 1), sofa (row 2, row 3), motorbike (row 5) amnd cat (row3, row 5). Similar to PASCAL VOC 2012, we emphasize the following points: (1) We observe the proposed bias reduction loss predicts more unseen classes, for example, it adjusts the prediction from SPNet to the cow and cat classes in the first row and 3rd row respectively. (2) The superpixel helps our model recognize the unseen objects better and outputs the entire masks of the objects (i.e., the boat, cat, motorbike for the first, second and third row). Other such examples include motorbike and cat in fifth row, where superpixel pooling have helped improve the features learnt for the novel classes. It also has helped in smoothing over the object shape as can be seen in first row, where cow's mask is smoothed over its shape, compared to the prediction of SPNet + $\mathcal{L}_{BR}$. Similar for cat's mask prediction in third row, where such smoothing helps in improving the performance. (3) Our $(SP)^2$Net segments the background classes more smoothly on the inner regions and precisely on the boundaries. For eg, ground in third row and floor in fourth where both SPNet and SPNet + $\mathcal{L}_{BR}$ has incorrect predictions. On the other hand, our $(SP)^2$Net predicts these classes smoothly.

Consequently, we conclude our $(SP)^2$Net obtained significant improvements over the baseline SPNet [61] and ranked at today's state of the art performance for the task of generalized zero-label semantic segmentation according to a series of qualitative and quantitative comparison to the baseline method [61] and other recent approaches [31, 6].

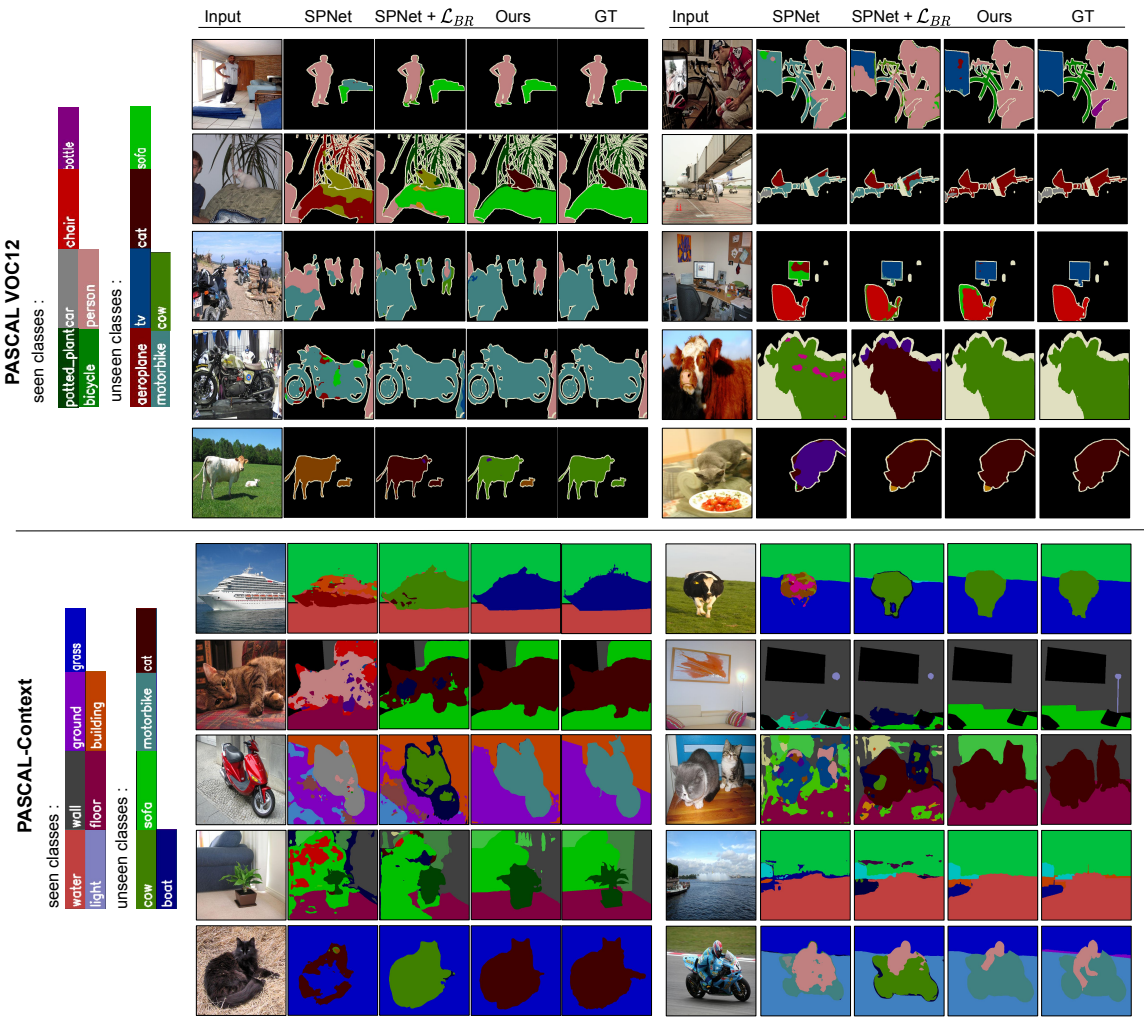


Figure 5-3: Qualitative comparisons with SPNet [61] and SPNet + $\mathcal{L}_{BR}$ on PASCAL VOC 2012 and PASCAL-Context datasets under the U-6 setting. Black color indicates ignore background regions.

Chapter 6

Conclusions, Summary and Future Works

We present a novel approach $(SP)^2$Net for the challenging generalized zero-label semantic segmentation task. The main novelties lie in the superpixel pooling module that aggregates features from adaptive superpixel regions and an efficient bias reduction loss that minimizes the confidence of seen classes in the ignore regions. We empirically show that our superpixel pooling module significantly improves the generalization to novel classes and our bias reduction loss effectively alleviates the data imbalance issue. We benchmark our $(SP)^2$Net against various baselines on 10 different splits from two datasets and establish a new state of the art on all data splits in GZLSS.

6.1 Summary.

In this thesis, first we introduced the problem of Generalised Zero Shot Semantic Segmentation. The goal of the task is to predict pixel labels which can be both seen or novel at evaluation. GZLSS task is challenging as the trained network needs to predict the pixel label for both seen and novel classes, while being trained on only seen class labels. The trained networks are usually biased towards predicting seen class labels. Semantic segmentation task requires pixel level annotation which is expensive, and solving GZLSS task helps in reducing this annotation effort. Also, the data distribution in real word setting is long tailed. Getting pixel level annotation for rare (tail-end classes) is difficult. GZLSS caters to solving semantic segmentation for such classes.

We further discuss existing approaches to solve GZLSS task. Existing approaches to solve this task either suffer from the seen label biased prediction issue or are unable to produce quality visual representations that generalises well to novel classes. Our proposed $(SP)^2$Net aims to solve these two important issues for GZLSS task. Next, we also discuss other related works in the domain. Specifically we discuss existing work on Zero-Shot learning, Few-shot semantic segmentation and superpixel guided

semantic segmentation. We show how existing work is different compared to our work. After we discuss, existing SPNet model. We use SPNet as our baseline model, which uses semantic projection layer to compute the compatibility of pixel features and class label embeddings. We also discuss, the data imbalance issue for SPNet model, which results in seen label biased predictions.

Next, we present our $(SP)^2$Net for GZLSS task. We extend SPNet by introducing superpixel pooling module and an additional bais reduction loss. It consists of three major segments, namely superpixel pooling module, semanitc projection module, bias reduction loss. The superpixel pooling module is applied at the head of the backbone segmentation network (eg DeepLab-v3+). It aggregates features from adaptive superpixel regions which helps in learning representations that generalises well on the unseen classes. Our superpixel pooling is very simple and doesn't require any parameter. We simply average the pixel features of a given superpixel region and replace each pixel feature with the averaged feature value. It is able to provide boundary information along with contextual information as priors to the network. Despite of its simplicity, this pooling method shows to improve generalisation on unseen classes. Superpixels are often noisy and pooling features in a such superpixels results in degradation of performance. To overcome training from such noisy signals, we presented superpixel correction, that makes sure we do superpixel pooling on correct superpixels only. Next, we introduce the bias reduction loss. Since, we train with no samples of unseen classes, the trained model is highly biased towards seen classes. The bias reduction loss helps in solving this biaseness towards seen classes by minimizing the confidence of seen classes in ignore regions. After applying bias reduction loss, we are able to solve seen label biased predictions.

Following, we present experiments to validate our claims. We evaluated our $(SP)^2$Net on two datasets PASCAL VOC 2012 and PASCAL-Context on 10 different datasplits. We outperformed the existing state of the art in most of the data-splits. We presented ablation study to study the imapact of superpixel pooling location and found pooling the output of segmentation head (ie before ASPP module in DeepLab-v3+) performs best. We also perform ablation study the impact of superpixel level (size) on model performance. We observe performance drop for coarser superpixels because of wrong pooling of noisy superpixels. Further to study the importance of superpixel pooling we also tried to change the segmentation backbone and found improvement in model performance with different backbone, suggesting our pooling module works with other segmentation backbones as well. We also provide qualitative images to show the qualitative improvement of our model compared to SPNet model, and SPNet model with bias reducetion loss.

6.2 Future Work.

Exploiting superpixel for GZLSS task is a novel approach and it opens further areas of improvement in the same GZLSS task. We discuss possible future work involving

superpixels and GZLSS below :

- **Pseudo label based improvement.** We can work on Pseudo label based improvement on our current work. The pseudo labels obtained for the unseen classes by the current model can further be used as training samples for unseen classes. In order to provide consistency for the pseudo labels, we can use majority voting for a label in a given superpixel.

- **Superpixel based Self Supervision.** In the current work we focussed on improving the representations learnt, that could generalise well on the unseen classes. We can further work on this improvement with additional superpixel based self supervision loss during training. The core idea being, the features belonging to same superpixel should be near in representation space compared to features belonging to different superpixels. This idea can be exploited to improve on the existing learned representations.

- **Superpixel Ensemble method.** Currently, the $(SP)^2$Net model uses a fixed defined (corrected) superpixel. But having a predefined superpixel for pooling is suboptimal. Instead the model should be able to give weight to different available superpixels for a given region of interest and do weighted pooling. This would remove the requirement of superpixel correction step and improve performance.

- **Uncertainity minimisation for transfer learning.** The SPNet method uses transfer learning from seen to unseen classes for GZLSS task. The pixel features belonging to class boundaries (or non-discriminative regions) are uncertain about the class prediction and thus learning from such pixels is not optimal. We should reduce this uncertainity in learning. Our $(SP)^2$Net method helps in performing superpixel pooling for entire superpixel region, thus reducing this uncertainity. But with current average pooling, we provide equal wieghtage to all pixel features in a superpixel, which is not correct. With adaptive weighted average pooling with learnable weights, we can further solve this issue.

Appendix A

More Qualitative Results

We provide qualitative results for different splits (split-2 - Figure A-1, split-4 - Figure A-2, split-6 - Figure A-3, split-8 - figure A-4 and split-10 - Figure A-5) for both PASCAL VOC 2012 and PASCAL-Context datasets. From those figures, we can observe our final model achieves significant improvements compared to the baseline SPNet. For example, even though there are only two unseen classes in Figure A-1, SPNet is affected and omits unstable and non-smooth results on unseen classes motorbike or cow. Besides, we can also clearly see the benefits of applying our bias reduction loss, which adjusts many regions from seen classes to unseen. For example, in the 3$^{\text{rd}}$ row of Figure A-2 for PASCAL VOC 2012, SPNet only predicts a small portion of the sofa, even it does not have a complex combination of various colors. In contrast, the SPNet with our new loss and our final model are able to segment the sofa successfully. Last but not least, we also find our model with superpixels produces much better boundaries compared to SPNet and SPNet+$\mathcal{L}_{BR}$ in both datasets and splits.

Consequently, we demonstrate the effectiveness of our proposed model and two components on generalized zero-label semantic segmentation.

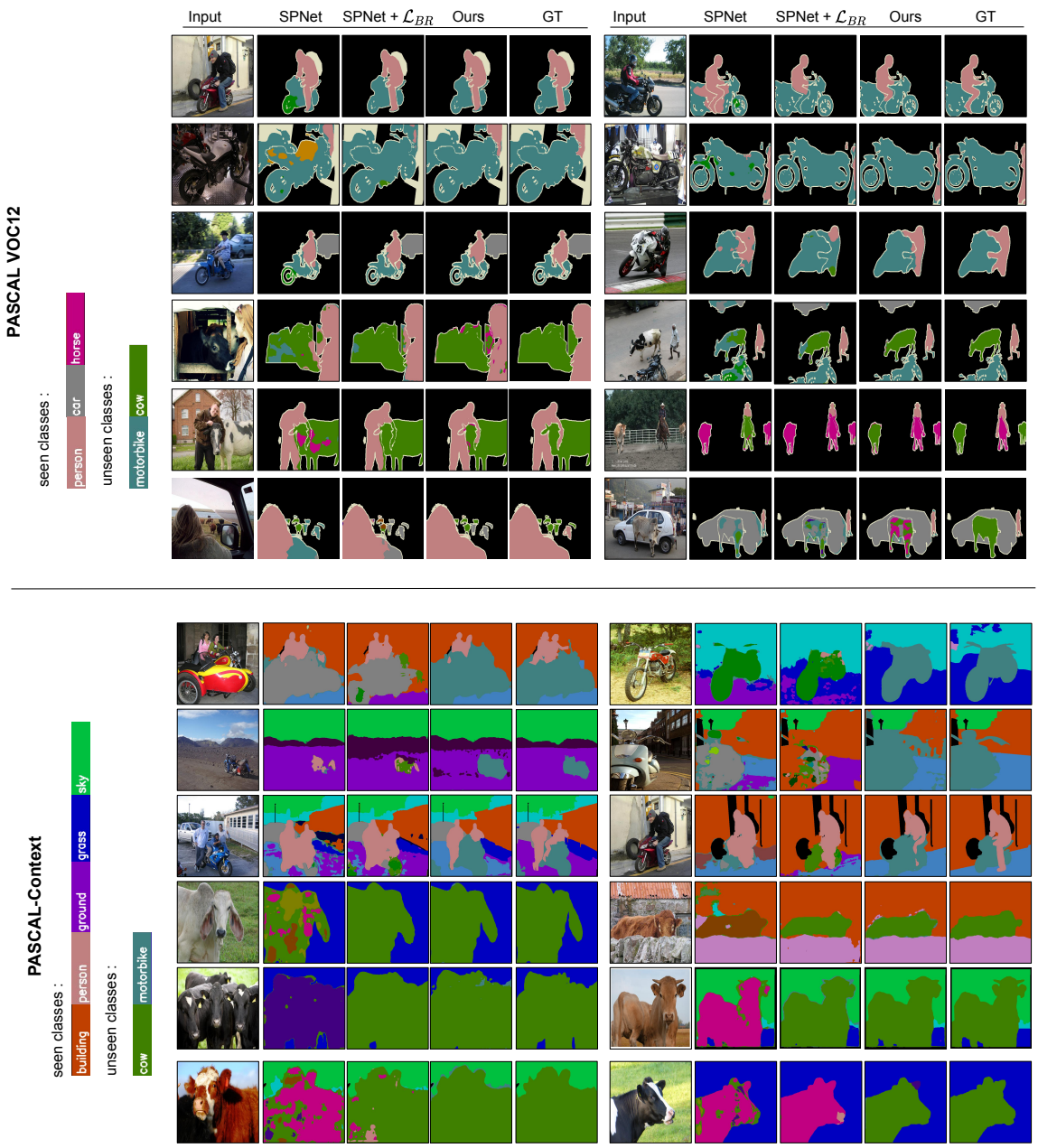


Figure A-1: Qualitative comparisons with SPNet and SPNet + $\mathcal{L}_{BR}$ on PASCAL VOC 2012 and PASCAL-Context datasets under the Unseen-2 setting. Black color indicates ignore background regions.

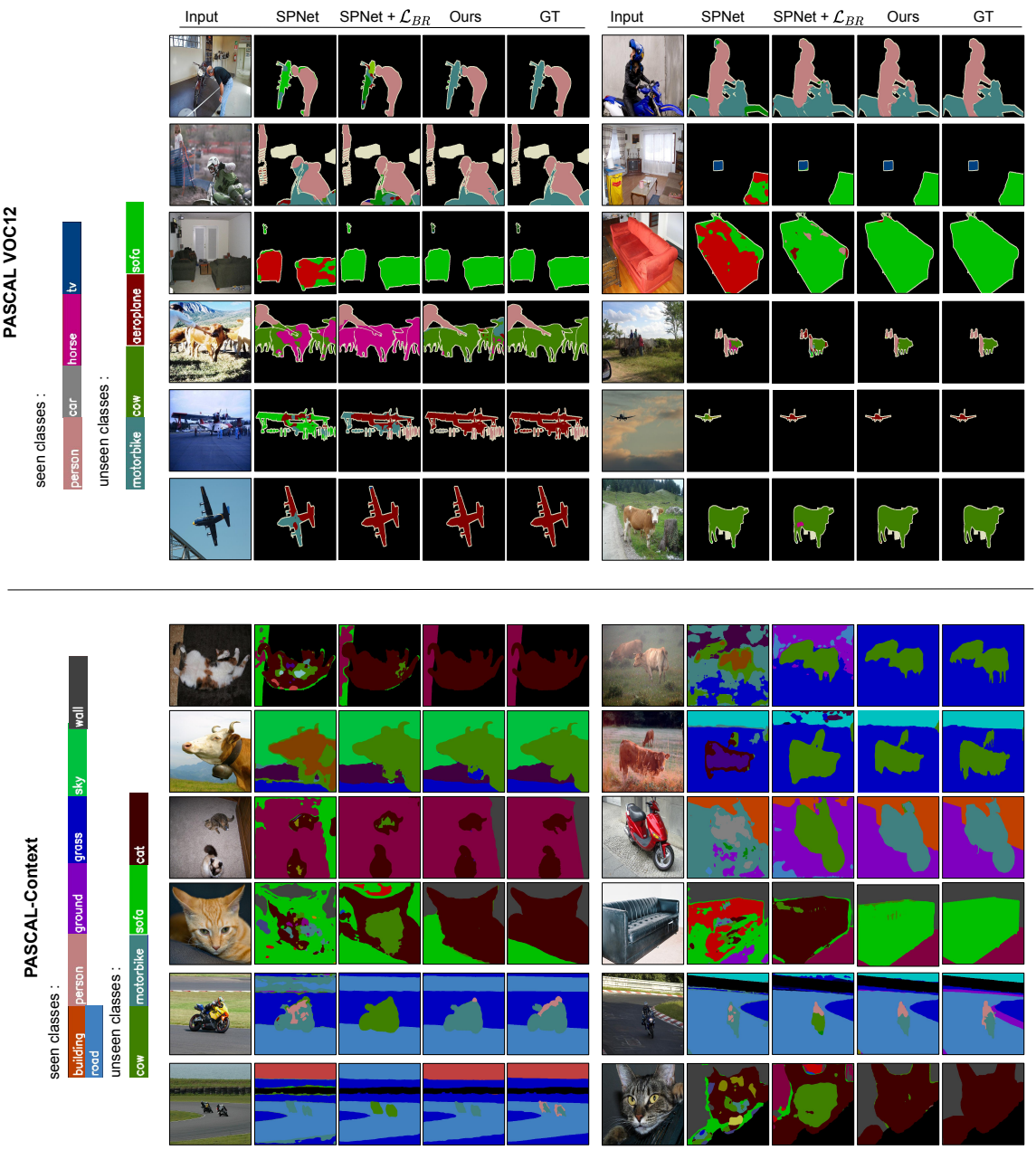


Figure A-2: Qualitative comparisons with SPNet and SPNet + $\mathcal{L}_{BR}$ on PASCAL VOC 2012 and PASCAL-Context datasets under the Unseen-4 setting. Black color indicates ignore background regions.

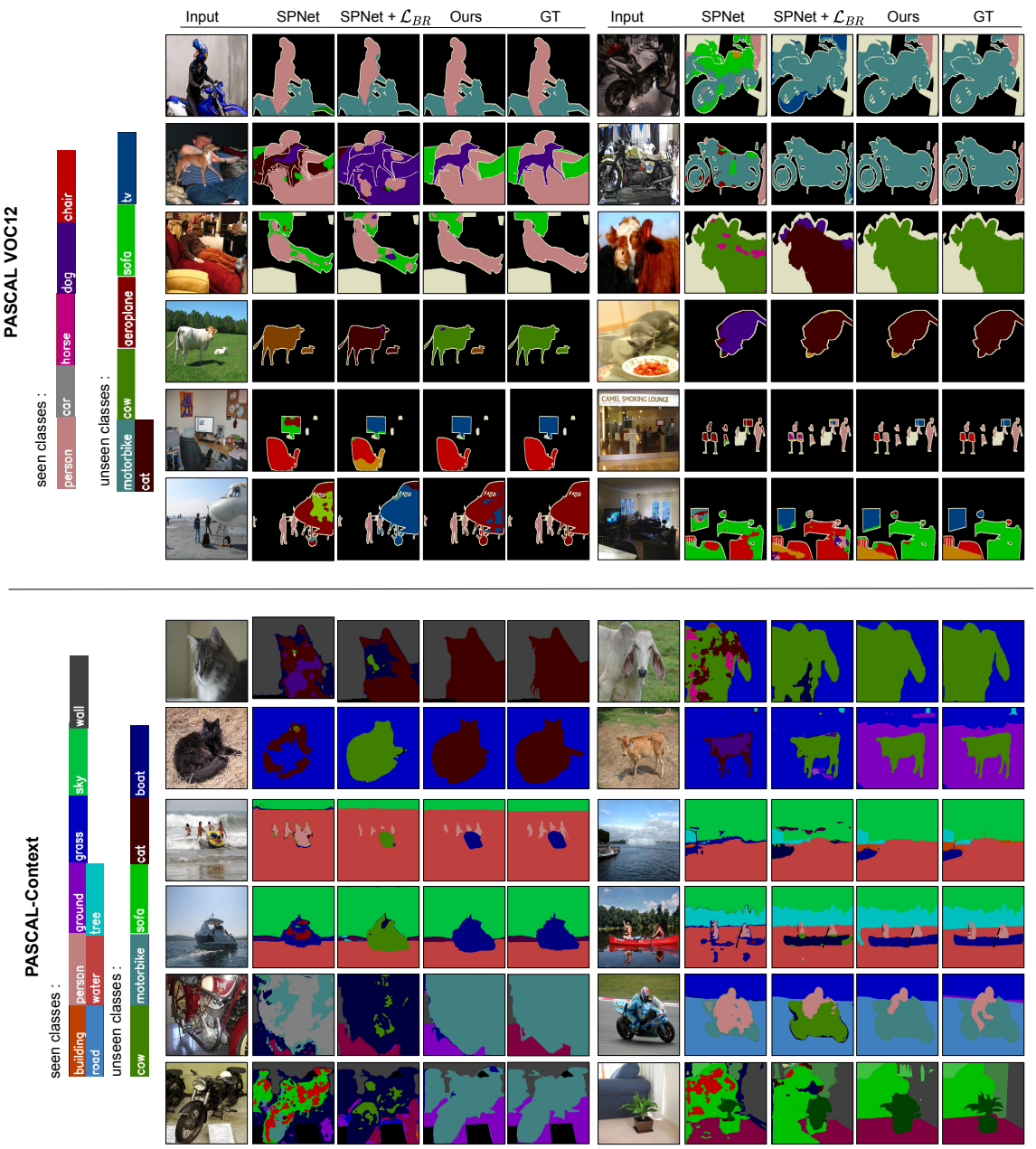


Figure A-3: Qualitative comparisons with SPNet and SPNet + $\mathcal{L}_{BR}$ on PASCAL VOC 2012 and PASCAL-Context datasets under the Unseen-6 setting. Black color indicates ignore background regions.

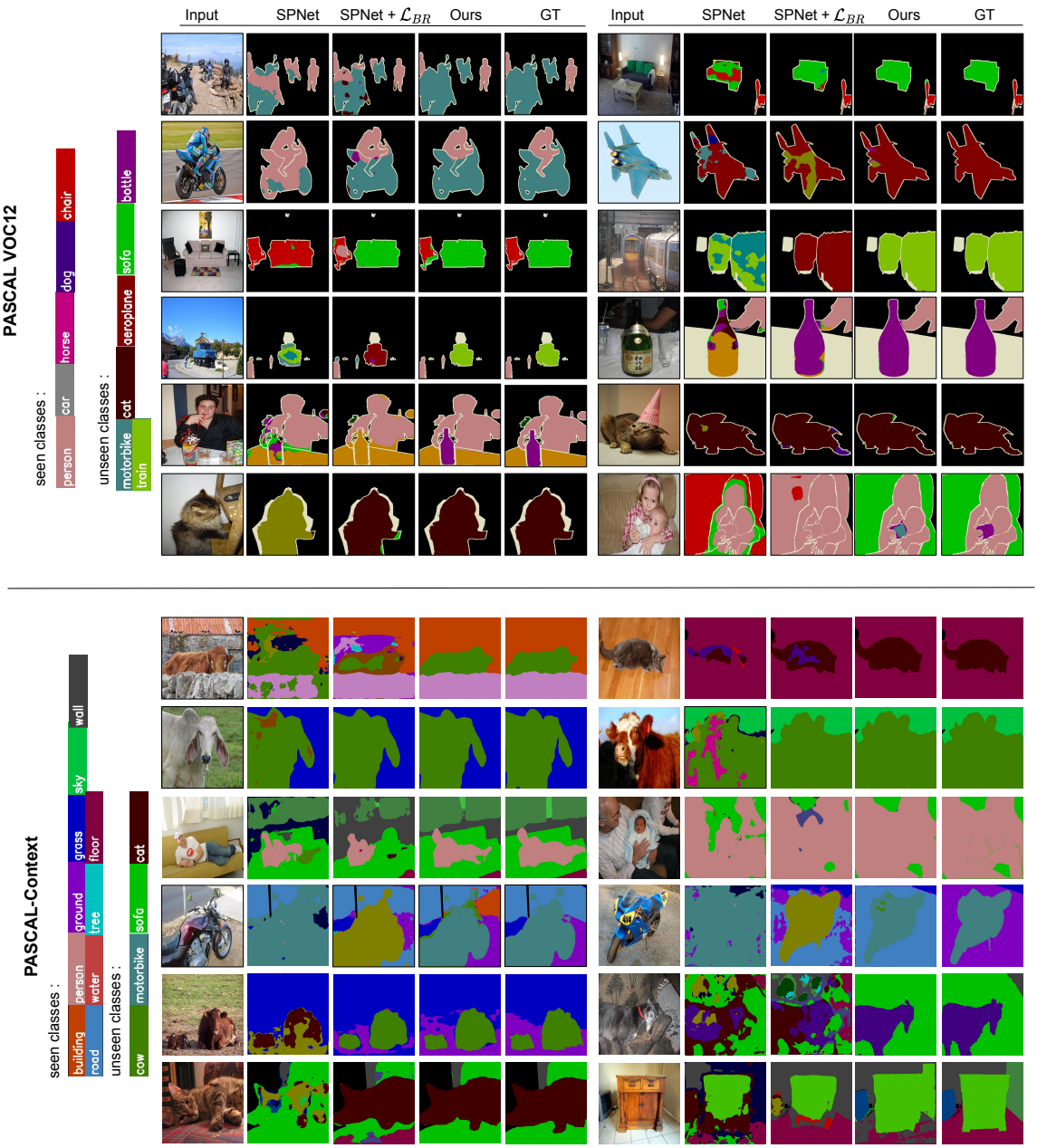


Figure A-4: Qualitative comparisons with SPNet and SPNet + $\mathcal{L}_{BR}$ on PASCAL VOC 2012 and PASCAL-Context datasets under the Unseen-8 setting. Black color indicates ignore background regions.

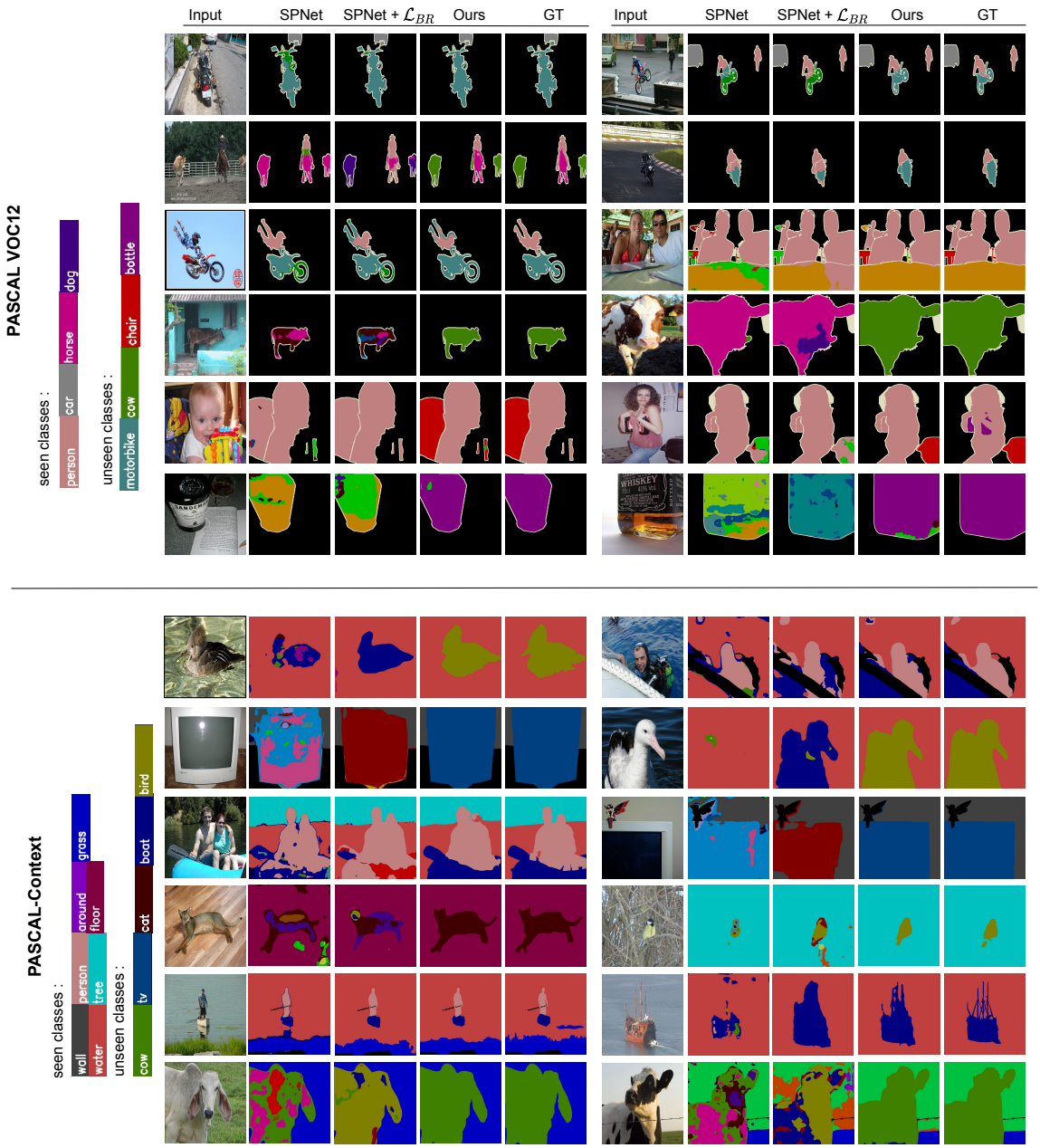


Figure A-5: Qualitative comparisons with SPNet and SPNet + $\mathcal{L}_{BR}$ on PASCAL VOC 2012 and PASCAL-Context datasets under the Unseen-10 setting. Black color indicates ignore background regions.

Bibliography

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels. Technical report, 2010. (cited on page 21)

[2] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 2012. (cited on page 20, 21, 28)

[3] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *TPAMI*, 2016. (cited on page 25)

[4] Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 328–335, 2014. (cited on page 13, 20, 21, 28)

[5] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *ECCV*, 2016. (cited on page 11)

[6] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. In *NeurIPS*, 2019. (cited on page 12, 14, 27, 29, 32, 35, 36, 37, 44)

[7] Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *ECCV*. Springer, 2016. (cited on page 25)

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2017. (cited on page 27, 28, 35, 36)

[9] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. (cited on page 12, 13, 17, 19, 22, 28, 29, 35, 36, 38, 39, 40)

[10] Subhabrata Chowdhury. Semantic Projection Network for Zero- and Few-label Semantic Segmentation. Master's thesis, Universität des Saarlandes, Saarbrücken, 2019. (cited on page 13)

[11] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. (cited on page 35)

[12] Andrea Frome, Greg Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NeurIPS*, 2013. (cited on page 25)

[13] Brian Fulkerson, Andrea Vedaldi, and Stefano Soatto. Class segmentation and object localization with superpixel neighborhoods. In *2009 IEEE 12th international conference on computer vision*, pages 670–677. IEEE, 2009. (cited on page 21)

[14] Raghudeep Gadde, Varun Jampani, Martin Kiefel, Daniel Kappler, and Peter V Gehler. Superpixel convolutional networks using bilateral inceptions. In *ECCV*, 2016. (cited on page 28)

[15] Stephen Gould, Jim Rodgers, David Cohen, Gal Elidan, and Daphne Koller. Multi-class segmentation with relative location prior. *International Journal of Computer Vision*, 80(3):300–316, 2008. (cited on page 21)

[16] Leo Grady. Random walks for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 28(11):1768–1783, 2006. (cited on page 21)

[17] Leo Grady and Gareth Funka-Lea. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. In *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis*, pages 230–245. Springer, 2004. (cited on page 21)

[18] Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. Context-aware feature generation for zero-shot semantic segmentation. In *ACM Multimedia*, 2020. (cited on page 12)

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. (cited on page 35)

[20] Shengfeng He, Rynson WH Lau, Wenxi Liu, Zhe Huang, and Qingxiong Yang. Supercnn: A superpixelwise convolutional neural network for salient object detection. *International journal of computer vision*, 115(3):330–344, 2015. (cited on page 21)

[21] Yang He, Wei-Chen Chiu, Margret Keuper, and Mario Fritz. Std2p: Rgbd semantic segmentation using spatio-temporal data-driven pooling. In *CVPR*, 2017. (cited on page 28)

[22] Ping Hu, Stan Sclaroff, and Kate Saenko. Uncertainty-aware learning for zero-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 33, 2020. (cited on page 27)

[23] Dinesh Jayaraman and Kristen Grauman. Zero-shot recognition with unreliable attributes. In *NIPS*, 2014. (cited on page 25)

[24] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016. (cited on page 35)

[25] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *CVPR*, 2017. (cited on page 11, 28)

[26] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24:109–117, 2011. (cited on page 18)

[27] Vinay Kumar Verma, Gundeep Arora, Ashish Mishra, and Piyush Rai. Generalized zero-shot learning via synthesized examples. In *CVPR*, 2018. (cited on page 25, 26)

[28] Suha Kwak, Seunghoon Hong, and Bohyung Han. Weakly supervised semantic segmentation using superpixel pooling network. In *AAAI*, 2017. (cited on page 12, 28)

[29] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009. (cited on page 25)

[30] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE transactions on pattern analysis and machine intelligence*, 36(3):453–465, 2013. (cited on page 25)

[31] Peike Li, Yunchao Wei, and Yi Yang. Consistent structural relation learning for zero-shot segmentation. In *NeurIPS*, 2020. (cited on page 12, 14, 27, 29, 32, 35, 36, 37, 44)

[32] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, 2016. (cited on page 12)

[33] Di Lin, Yuanfeng Ji, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Multi-scale context intertwining for semantic segmentation. In *ECCV*, 2018. (cited on page 28)

[34] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5162–5170, 2015. (cited on page 21)

[35] Miaomiao Liu, Mathieu Salzmann, and Xuming He. Discrete-continuous depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2014. (cited on page 21)

[36] Wei Liu, Andrew Rabinovich, and Alexander C Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015. (cited on page 30)

[37] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu. Large-scale long-tailed recognition in an open world. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. (cited on page 12)

[38] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. (cited on page 17, 18, 28)

[39] Vaïa Machairas, Etienne Decencière, and Thomas Walter. Waterpixels: Superpixels based on the watershed transformation. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 4343–4347. IEEE, 2014. (cited on page 21)

[40] Vaïa Machairas, Matthieu Faessel, David Cárdenas-Peña, Théodore Chabardes, Thomas Walter, and Etienne Decencière. Waterpixels. *IEEE Transactions on Image Processing*, 24(11):3707–3716, 2015. (cited on page 21)

[41] Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Pablo Arbeláez, and Luc Van Gool. Convolutional oriented boundaries: From image segmentation to high-level tasks. *IEEE TPAMI*, 2017. (cited on page 13, 20, 21, 27, 28, 35, 43)

[42] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. (cited on page 22, 27, 35)

[43] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013. (cited on page 25)

[44] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 891–898, 2014. (cited on page 35)

[45] Peer Neubert and Peter Protzel. Compact watershed and preemptive slic: On improving trade-offs of superpixel segmentation algorithms. In *2014 22nd international conference on pattern recognition*, pages 996–1001. IEEE, 2014. (cited on page 21)

[46] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013. (cited on page 25)

[47] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *2012 IEEE conference on computer vision and pattern recognition*, pages 733–740. IEEE, 2012. (cited on page 21)

[48] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2016. (cited on page 28)

[49] Xiaofeng Ren and Jitendra Malik. Learning a classification model for segmentation. In *Computer Vision, IEEE International Conference on*, volume 2, pages 10–10. IEEE Computer Society, 2003. (cited on page 21)

[50] Bernardino Romera-Paredes, ENG OX, and Philip HS Torr. An embarrassingly simple approach to zero-shot learning. In *ICML*, 2015. (cited on page 25)

[51] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *CVPR*, 2019. (cited on page 25, 26)

[52] Guang Shu, Afshin Dehghan, and Mubarak Shah. Improving an object detector and extracting regions using superpixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3721–3727, 2013. (cited on page 21)

[53] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017. (cited on page 28)

[54] Jie Song, Chengchao Shen, Yezhou Yang, Yang Liu, and Mingli Song. Transductive unbiased embedding for zero-shot learning. In *CVPR*, 2018. (cited on page 25, 26)

[55] David Stutz, Alexander Hermans, and Bastian Leibe. Superpixels: An evaluation of the state-of-the-art. *CVIU*, 2018. (cited on page 20, 28)

[56] Zhuotao Tian, Xin Lai, Li Jiang, Michelle Shu, Hengshuang Zhao, and Jiaya Jia. Generalized few-shot semantic segmentation. *arXiv preprint arXiv:2010.05210*, 2020. (cited on page 28)

[57] Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *TPAMI*, 2020. (cited on page 28)

[58] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NIPS*, 2016. (cited on page 28)

[59] David Weikersdorfer, Alexander Schick, and Daniel Cremers. Depth-adaptive supervoxels for rgb-d video segmentation. In *2013 IEEE International Conference on Image Processing*, pages 2708–2712. IEEE, 2013. (cited on page 21)

[60] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *CVPR*, 2016. (cited on page 25)

[61] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *CVPR*, pages 8256–8265, 2019. (cited on page 12, 14, 20, 22, 23, 27, 28, 29, 30, 31, 32, 35, 36, 37, 41, 44, 45)

[62] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, 2018. (cited on page 25, 26)

[63] Junjie Yan, Yinan Yu, Xiangyu Zhu, Zhen Lei, and Stan Z Li. Object detection by labeling superpixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5107–5116, 2015. (cited on page 21)

[64] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *CVPR*, 2017. (cited on page 25)

[65] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Thomas S Huang. Sg-one: Similarity guidance network for one-shot semantic segmentation. *IEEE Transactions on Cybernetics*. (cited on page 28)

[66] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. (cited on page 13, 17, 20, 28, 29, 40)

[67] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017. (cited on page 30)

[68] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. (cited on page 18)

[69] Yizhe Zhu, Jianwen Xie, Bingchen Liu, and Ahmed Elgammal. Learning feature-to-feature translator by alternating back-propagation for generative zero-shot learning. In *ICCV*, 2019. (cited on page 25)