UNIVERSITY OF CALIFORNIA, MERCED

Weakly-Supervised Semantic Segmentation via Self-Regularization

A thesis submitted in partial satisfaction of the
requirements for the degree
Master of Science

in

Electrical Engineering and Computer Science

by

Yu-Ting Chang

Committee in charge:

Professor Ming-Hsuan Yang, Chair
Professor Shawn Newsam
Professor Sungjin Im

2020

The thesis of Yu-Ting Chang is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Professor Shawn Newsam

Professor Sungjin Im

Professor Ming-Hsuan Yang Chair

University of California, Merced

2020

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

VITA

2014	B. E. in Electrical and Computer Engineering, National Chaio Tung University, Hsinchu, Taiwan
2020	M. S. in Electrical Engineering and Computer Science, University of California, Merced

PUBLICATIONS

Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, Ming-Hsuan Yang, *Weakly-Supervised Semantic Segmentation via Sub-category Exploration*, In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, Ming-Hsuan Yang, *Mixup-CAM: Weakly-supervised SemanticSegmentation via Uncertainty Regularization*, In British Machine Vision Conference (BMVC), 2020.

ABSTRACT OF THE THESIS

Weakly-Supervised Semantic Segmentation via Self-Regularization

by

Yu-Ting Chang
Master of Science in Electrical Engineering and Computer Science
University of California Merced, 2020
Professor Ming-Hsuan Yang, Chair

The goal of semantic segmentation is to assign a semantic category to each pixel in the image. It has been one of the most important tasks in computer vision that enjoys a wide range of applications such as image editing and scene understanding. Recently, deep convolutional neural network (CNN) based methods have been developed for semantic segmentation and achieved significant progress. However, such approaches rely on learning supervised models that require pixel-wise annotations, which take extensive effort and time. To reduce the effort in annotating pixel-wise ground truth labels, numerous weakly-supervised methods are proposed using various types of labels such as image-level, bounding box, point-level, and scribble-based labels. In this thesis, we focus on using image-level labels which can be obtained effortlessly, yet a more challenging case under the weakly-supervised setting.

Existing weakly-supervised semantic segmentation methods using image-level annotations typically rely on initial responses to locate object regions. However, such response maps generated by the classification network usually focus on discriminative object parts, due to the fact that the network does not need the entire object for optimizing the objective function. To address this issue, we improve the generated response map by enforcing the network to pay attention to other parts of an object via self-regularization techniques. First, we apply the mixup data augmentation to effectively calibrate the model uncertainty on overconfident predictions, which enables the model to attend to more object regions. Second, we introduce a self-supervised task

that discovers sub-categories in an unsupervised manner. By imposing a more challenging task, the model learns better representations, thereby improving the response map. Based on the proposed two self-regularization methods, the produced initial responses are more complete and balanced across object regions, which facilitates the latter steps for weakly-supervised semantic segmentation, i.e., response refinement and segmentation model training.

Chapter 1

Introduction

Semantic segmentation is one of the fundamental tasks in computer vision, with a wide range of applications such as image editing and scene understanding. In order to obtain reliable models and achieve promising performance, recently deep neural network (DNN) based methods [24, 6, 59] are learned from fully-supervised data that requires pixel-wise semantic annotations. However, acquiring such pixel-wise annotations is usually time-consuming and labor-intensive, which limits the application potentials in the real world. As a result, numerous approaches tackle this issue via training models only on weakly-annotated data, *e.g.*, image-level [1, 31, 41, 45], bounding box [40, 12, 29], point-level [2], scribble-based [36, 54], or video-level [8, 64, 51] labels. In this thesis, we focus on utilizing the image-level label, which is the most efficient scheme for weak annotations but also a challenging scenario.

Existing weakly-supervised semantic segmentation (WSSS) algorithms using image-level labels mainly consist of three steps: 1) localizing objects via a categorical response map, 2) refining the response map to generate pseudo annotations, and 3) training the semantic segmentation network using pseudo ground truths. Recent methods [1, 26, 55, 57] have achieved significant progress for WSSS, but most of them focus on improving the latter two steps. Since the success of these sequential steps hinges on the quality of the initial response map generated in the first step, in this thesis we present two effective self-regularization solutions that can generate better response maps to localize objects.

One common practice to generate the initial response map is using class activation

map (CAM) [65]. However, since CAM is typically supervised by a classification loss that could be sufficiently optimized through seeing only a small portion of objects, the generated response map usually only attends to partial regions. In this thesis, we aim to improve the class activation maps by enhancing feature representations in the classification network. We first propose the Mixup-CAM framework that calibrates the uncertainty in network prediction. Next, we introduce a self-supervised task to discover sub-categories in an unsupervised manner.

In Chapter 2, we present a principled and end-to-end trainable framework to allow the network to pay attention to other parts of the object, while producing a more complete and uniform response map. In particular, we introduce the mixup data augmentation scheme and integrate it into the classification network. Then we design two uncertainty regularization terms to better interact with the mixup strategy. In experimental results, we provide comprehensive analysis of each component in the proposed method and show that our approach achieves state-of-the-art performance against existing algorithms.

In Chapter 3, we propose a simple yet effective approach that introduces a self-supervised task by exploiting the sub-category information, which forces the network to pay attention to other parts of an object. Specifically, we perform clustering on image features to generate pseudo sub-categories labels within each annotated parent class, and construct a sub-category objective to assign the network a more challenging task. By iteratively clustering image features and learning the sub-category objective, the training process does not limit itself to the most discriminative object parts, hence improving the quality of the response maps. We conduct extensive analysis to validate the proposed method and show that our approach performs favorably against the state-of-the-art approaches. We conclude the thesis in Chapter 4 and discuss potential future directions.

Chapter 2

Mixup-CAM: Weakly-supervised Semantic Segmentation via Uncertainty Regularization

2.1 Introduction

Semantic segmentation is one of the fundamental tasks in computer vision, with a wide range of applications such as image editing and scene understanding. In order to obtain reliable models and achieve promising performance, recently deep neural network (DNN) based methods [24, 6, 59] are learned from fully-supervised data that requires pixel-wise semantic annotations. However, acquiring such pixel-wise annotations is usually time-consuming and labor-intensive, which limits the application potentials in the real world. As a result, numerous approaches tackle this issue via training models only on weakly-annotated data, *e.g.*, image-level [1, 31, 41, 45], bounding box [40, 12, 29], point-level [2], scribble-based [36, 54], or video-level [8, 64, 51] labels. In this thesis, we focus on utilizing the image-level label, which is the most efficient scheme for weak annotations but also a challenging scenario.

Existing weakly-supervised semantic segmentation (WSSS) algorithms mainly operate with three steps: 1) localizing objects via a categorical response map, 2) refining the response map to generate pseudo annotations, and 3) training the semantic segmenta-

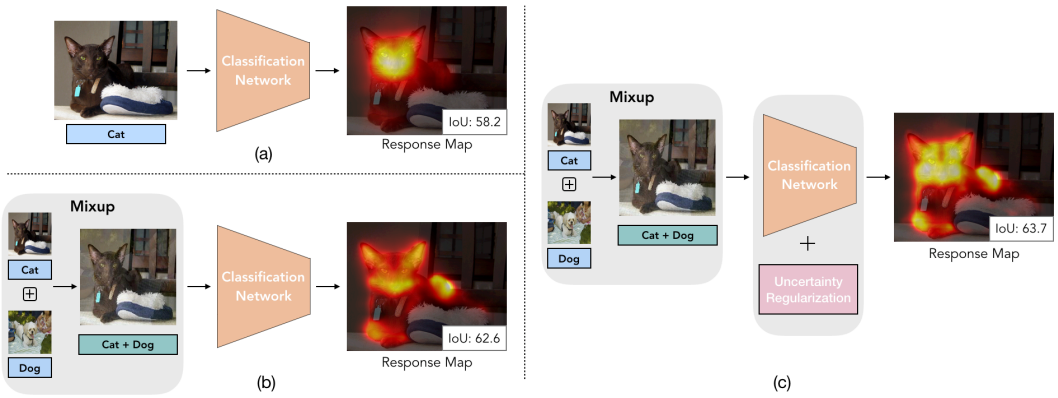

Figure 2.1: Comparisons of (a) the original CAM method; (b) CAM + the mixup data augmentation; and (c) the proposed Mixup-CAM framework that integrates the mixup scheme and the uncertainty regularization. Compared to (a) and (b), our final response map (c) attends to other object parts with more uniformly distributed response.

tion network using pseudo ground truths. Recent methods [1, 26, 55, 57] have achieved significant progress for WSSS, but most of them focus on improving the latter two steps. Since the success of these sequential steps hinges on the quality of the initial response map generated in the first step, in this chapter we present an effective solution to localize objects.

One common practice to produce the initial response map is using class activation map (CAM) [65]. However, since CAM is typically supervised by a classification loss that could be sufficiently optimized through seeing only a small portion of objects, the generated response map usually only attends on partial regions (see Figure 2.1(a)). To tackle this issue, recent methods [33, 28] make efforts to improve the response map via using the dropout strategy that increases the model uncertainty or aggregating maps produced at different stages to see more object parts. However, there remains a challenge whether there are better loss function designed to explicitly facilitate the model training and produce better response maps, which are not addressed in prior works.

In this chapter, we propose a principled and end-to-end trainable network with loss functions designed to systematically control the generation of the response map. First, inspired by the mixup data augmentation in [62], we observe that including mixup could effectively calibrate the model uncertainty on overconfident predictions [50] and in re-

turn enables the model to attend to more object regions. However, it is challenging to control the mixup augmentation process and the model uncertainty, due to non-uniform response distributions (see Figure 2.1(b)), which may affect subsequent response refinement steps. Therefore, we introduce another two loss terms to the mixup process by regularizing the class-wise uncertainty and the spatial response distribution. We refer to our model as *Mixup-CAM* and show that the produced response map is more complete and balanced across object regions (see Figure 2.1(c)), which facilitates the latter response refinement and segmentation model training steps.

We conduct quantitative and qualitative experiments to demonstrate the effectiveness of the proposed Mixup-CAM method on the PASCAL VOC 2012 dataset [16]. To the best of our knowledge, our algorithm is the first to demonstrate that mixup could improve the WSSS task on complicated multi-labeled images, along with other designed loss functions to produce better response maps. In addition, we present the ablation study and more analysis to validate the importance of each designed loss. Finally, we show that our method achieves state-of-the-art semantic segmentation performance against existing approaches.

2.2 Related Work

Initial Prediction for WSSS. Initial cues are essential for segmentation tasks since they provide reliable priors to generate segmentation maps. The class activation map (CAM) [65] is a common practice for localizing objects. It highlights class-specific regions that serve as the initial cues. However, since the CAM model is trained by a classification network, it tends to attend to small discriminative parts of the object, leading to incomplete initial masks. Several methods have been developed to alleviate this problem. Approaches like [48, 56, 63, 34] deliberately hide or erase the regions of an object, forcing the model to look for more diverse parts. However, such strategies require iterative model training and response aggregation steps. After gradual expansion of the attention regions, non-object regions are prone to be activated, which leads to inaccurate attention maps. Other algorithms [25, 61] use both object and background cues to prevent the attention map from including more background regions, yet pixel-

level saliency labels are used.

Instead of using the erasing scheme, recently the FickleNet approach [33] introduces the stochastic feature selection to obtain a diverse combination of locations on feature maps. Moreover, the OAA method [28] adopts an online attention accumulation strategy to collect various object parts discovered at different training stages. By aggregating the attention maps, it could obtain an initial cue that contains a larger region of the object. Unlike methods that mitigate the problem by discovering complementary regions via iterative erasing steps or consolidating attention maps, our proposed approach aims at harnessing the uncertainty in end-to-end classification learning. In addition, by regularizing both class-wise uncertainties and spatial response distributions, our approach averts the attention from focusing on small parts of the semantic objects, hence producing much improved response maps.

Response Refinement for WSSS. Various approaches [1, 17, 18, 26, 31, 55, 57] are proposed to refine the initial cue by expanding the region of attention maps. Other methods [1, 17, 18] are developed using affinity learning. The recent SSDD scheme [47] proposes a difference detection module to estimate and reduce the gap between the initial mask and final segmentation results in a self-supervised manner. However, the performance of these methods is limited as initial seeds are still obtained from CAM-like methods. If these seeds only come from the discriminative parts of the object, it is difficult to expand regions into non-discriminative parts. Moreover, the initial prediction may produce wrong attention regions, which would lead to even more inaccurate regions in subsequent refinement steps.

Label-preserving *vs.* Non-preserving Augmentations. Data augmentation is a common regularization technique in both supervised and unsupervised learning [11, 35, 22]. Conventional data augmentation techniques such as scaling, rotation, cropping, color augmentation, and Gaussian noise can change the pixel values of an image without altering its labels. These label-preserving transformations are commonly applied in training deep neural networks to improve the model generalization capabilities.

Recent work has demonstrated that even non-label-preserving data augmentation can be surprisingly effective. Explicit label smoothing has been adopted successfully to improve the performance of deep neural models. The Mixup method [62] is proposed

to train a neural network on a convex combination space of image pairs and their corresponding labels. It has been proven effective for the classification task and increases the robustness of neural networks. Numerous Mixup variants [3, 49, 60, 53, 50, 20] have been proposed to extend mixup for better prediction of uncertainty and calibration of the DNNs. These methods exhibit shared similarity of producing better-generalized models.

Entropy Regularization. Aside from mixup schemes for predictive uncertainty, another common uncertainty measure is entropy, which could act as a strong regularizer in both supervised and semi-supervised learning [44, 19]. In particular, [44] discourages the neural network from being over-confident by penalizing low-entropy distributions, while [19] utilizes entropy minimization in a semi-supervised setting as a training signal on unlabeled data. In this chapter, we also adopt the entropy-based loss to regularize the uncertainty, coupled with the mixup data augmentation for producing better response maps on objects.

2.3 Proposed Algorithm

We first describe the overall algorithm and introduce details of the proposed Mixup-CAM framework with loss functions designed to improve the initial response map. We then detail how to generate the final semantic segmentation results.

2.3.1 Algorithm Overview

One typical way to generate response maps for annotated object categories is to use CAM [65]. However, these response maps tend to focus on discriminative object parts, which are less effective for the WSSS task. One reason is that CAM relies on the classification loss, which only requires partial object regions to be activated during training. As a result, when the objective is already optimized with high confidence, the model may not attempt to learn other object parts.

In this chapter, we propose to integrate the idea of mixup data augmentation [62], thereby calibrating the uncertainty in prediction [50] as well as allowing the model to

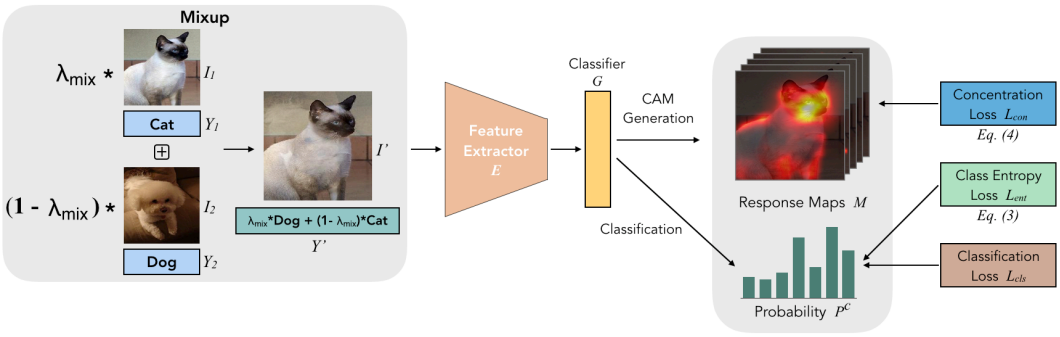

Figure 2.2: Overview of Mixup-CAM. We perform mixup data augmentation on input images with their corresponding labels via (2.2) and pass the mixed image through the feature extractor E and the classifier G to obtain the probability score P^c for each category c. For loss functions, in addition to the classification loss $\mathcal{L}_{cls}$ on mixup samples, we design two terms to regularize class-wise entropy ($\mathcal{L}_{ent}$ via (2.3)) and spatial distribution on CAM ($\mathcal{L}_{con}$ via (2.4)).

attend to other regions of the image. Although we find that adding mixup could improve the response map, sometimes the response could diverge too much, resulting in more false-positive object regions. To further regularize this uncertainty, we introduce two additional loss terms: the spatial loss and the class-wise loss. We illustrate the overall model and loss designs in Figure 2.2 and provide more details in the next subsection.

After receiving the initial response map, we utilize the method in [1] to expand and refine the response. Finally, we generate pseudo ground truths from the refined response and train a semantic segmentation network to obtain the final segmentation output. Note that while we focus on the first step of the initial response map in this thesis, the succeeding two steps could be replaced with alternative modules or models.

2.3.2 Mixup-CAM

CAM Generation. We first describe the CAM method for producing the initial response map as our baseline (see Figure 2.1(a)). The base network begins with a feature extractor E, followed by a global average pooling (GAP) layer and a fully-connected layer G as the output classifier. Next, given an input image I with its image-level labels Y, the network is trained with a multi-label classification loss $\mathcal{L}_{cls}(Y, G(E(I)))$ fol-

lowing [65]. After training this classification network, the activation map M^c for each category c is obtained by applying the c-channel classifier weight θ_G^c on the feature map $f = E(I)$:

$$M^c = \theta_G^{c\top} f. \tag{2.1}$$

Finally, the response is normalized by the maximum value of M^c.

Mixup Data Augmentation. Since the original classification network could easily obtain high confidence, in which the generated CAM only attends to small discriminative object parts, we utilize mixup data augmentation to calibrate the uncertainty in prediction [50]. Given an image pair $\{I_1, I_2\}$ and its label $\{Y_1, Y_2\}$ randomly sampled from the training set, we augment an image I' and its label Y' via:

$$\begin{aligned} I' &= \lambda_{mix} I_1 + (1 - \lambda_{mix}) I_2, \\ Y' &= \lambda_{mix} Y_1 + (1 - \lambda_{mix}) Y_2, \end{aligned} \tag{2.2}$$

where λ_{mix} is sampled from the *Beta*(α, α) distribution following [62]. Using this augmented data, we feed it into the classification network to minimize the loss $\mathcal{L}_{cls}(Y', G(E(I')))$ and follow the same procedure in (2.1) to produce the response map (see Figure 2.1(b)).

Compared to the original CAM generation, our network no longer receives a pure image but a mixed image that could have multiple objects with their weights based on λ_{mix} as in (2.2). Therefore, the predictive uncertainty could be enhanced, leading to smoother output distributions and enforcing the model to pay attention to other regions in the image in order to satisfy the classification loss $\mathcal{L}_{cls}(Y', G(E(I')))$.

Uncertainty Regularization. Although mixup could improve the response map by looking at other parts in the image, sometimes the response could become too divergent and thus attend to pixels non-uniformly, *e.g.*, Figure 2.1(b). This is attributed to the difficulty in controlling the quality of mixed images, especially when the model faces more complicated images such as PASCAL VOC, *e.g.*, an object could appear at various locations of the image with noisy background clutters.

To further facilitate the mixup process, we propose to self-regularize the uncertainty via class-wise loss and spatial loss terms. The first term is to directly minimize the

entropy in output prediction from the classifier to reduce uncertainty:

$$\mathcal{L}_{ent}(G(E(I'))) = -\frac{1}{HW}\sum_{h,w}\sum_{c\in C} P^c(h,w)\log P^c(h,w), \tag{2.3}$$

where C is the category number and $P^c \in \mathbb{R}^{H\times W}$ is the output probability for category c. Since our classifier G outputs multi-label probability, we concatenate the probabilities and normalize them by the maximum value, then calculate the final P with *softmax*.

Although the first term has the ability to minimize the uncertainty, it does not explicitly operate on the response map. To better balance the distribution on the response, we utilize a concentration loss similar to [27] and apply it directly on CAM for each category (*i.e.*, M^c), which encourages activated pixels to be spatially close to the response center:

$$\mathcal{L}_{con}(M) = \sum_{c\in\bar{C}}\sum_{h,w} ||\langle h,w\rangle - \langle \mu_h^c, \mu_w^c\rangle||^2 \cdot \hat{M}^c(h,w), \tag{2.4}$$

where $\mu_h^c = \sum_{h,w} h \cdot \hat{M}^c(h,w)$ is the center in height for category c (similarly for μ_w^c), $\hat{M}^c$ is the normalized response of M^c to represent a spatially distributed probability map. Note that, here we only calculate the concentration loss on presented categories $\bar{C}$ as provided in the image-label Y to avoid confusing the model with invalid categories.

Overall Objective. We have described our proposed Mixup-CAM framework, including mixup data augmentation in (2.2) and two regularization terms, *i.e.*, (2.3) and (2.4). To train the entire model in an end-to-end fashion, we perform the online mixup procedure and jointly optimize the following loss functions:

$$\mathcal{L}_{all} = \mathcal{L}_{cls}(I', Y') + \lambda_{ent}\mathcal{L}_{ent}(I') + \lambda_{con}\mathcal{L}_{con}(M). \tag{2.5}$$

For simplicity, we omit the detailed notation inside each loss term. We also note that M is produced online via computing (2.1) on valid categories in each forward iteration.

2.3.3 Implementation Details

Classification Network. Similar to [1], we use the ResNet-38 architecture [58] as our classification network, which consists of 38 convolution layers with wide channels, followed by a 3×3 convolution layer with 512 channels for better adaptation to the

classification task, a global average pooling layer, and two fully-connected layers for classification. In training, we adopt the pre-trained model on ImageNet [14] and finetune it on the PASCAL VOC 2012 dataset. Typical label-preserving data augmentations, *i.e.*, horizontal flip, random cropping, random scaling, and color jittering, are utilized on the training set.

We implement the proposed Mixup-CAM framework using PyTorch with a single Titan X GPU with 12 GB memory. For training the classification network, we use the Adam optimizer [30] with initial learning rate of 1e-3 and the weight decay of 5e-4. For mixup, we use $\alpha = 0.2$ in the *Beta*(α, α) distribution. For uncertainty regularization, we set λ_{ent} as 0.02 and λ_{con} as 2e-4. Unless specified otherwise, we use the same parameters in all the experiments. In the experimental section, we show studies for the sensitivity of different parameters.

Semantic Segmentation Generation. Based on the response map generated by our Mixup-CAM, we adopt the random walk approach via affinity [1] to refine the response and produce pixel-wise pseudo ground truths for semantic segmentation. In addition, similar to existing methods, we adopt dense conditional random fields (CRF) [32] to further refine the response and obtain better object boundaries. Finally, we utilize the Deeplab-v2 framework [6] with the ResNet-101 architecture [23] and train the segmentation network

2.4 Experimental Results

In this section, we present our main results of the proposed Mixup-CAM method for the WSSS task. First, we show that our approach achieves better initial response maps and further improves the subsequent refinement step. Second, we demonstrate the importance of each designed component. Finally, we provide evaluations on final semantic segmentation outputs in the PASCAL VOC dataset [16] against the state-of-the-art approaches. More results can be found in the supplementary material. We will make our code and models available to the public.

Table 2.1: IoU results of CAM and its refinement on the PASCAL VOC training set.

Method	CAM	CAM + Refinement
AffinityNet [1]	48.0	58.1
Mixup $\mathcal{L}_{cls}$	49.3	60.5
Mixup $\mathcal{L}_{cls}$ + $\mathcal{L}_{ent}$	49.5	61.6
Mixup $\mathcal{L}_{cls}$ + $\mathcal{L}_{con}$	49.9	61.7
Mixup $\mathcal{L}_{cls}$ + $\mathcal{L}_{ent}$ + $\mathcal{L}_{con}$	**50.1**	**61.9**

2.4.1 Evaluated Dataset and Metric

We conduct experiments on the PASCAL VOC 2012 semantic segmentation benchmark [16] with 21 categories, including one background class. Following existing WSSS methods, we use augmented 10,528 training images [21] to train our network. For evaluation of response maps of the training set, we use the set without augmentation with 1,464 examples, following the setting in [1]. For final semantic segmentation results, we use 1,449 images in the validation set to compare our results with other methods[1]. In all experiments, the mean Intersection-over-Union (mIoU) ratio is used as the evaluation metric.

2.4.2 Ablation Study and Analysis

Improvement on Response Map. We first present results of the initial and refined response maps. In Table 2.1, we show the performance for the original CAM used by the baseline AffinityNet [1], our CAM using the mixup data augmentation (Mixup $\mathcal{L}_{cls}$), and our final Mixup-CAM with mixup and uncertainty regularization (Mixup $\mathcal{L}_{cls}$ + $\mathcal{L}_{ent}$ + $\mathcal{L}_{con}$). In both results of CAM and its refinement, our IoU improvements are consistent after gradually adding the mixup augmentation and regularization. In addition, Figure 2.3 shows some example results of the initial response, which illustrates that our Mixup-

[1]Although there is a test set that can be evaluated on the official PASCAL VOC website, by the submission deadline the website is still out of service for returning the evaluated performance.

Figure 2.3: Sample results of initial responses. Our approach often produces the response map that covers more complete region of the object (i.e., attention on the body of the animal), while the initial cue obtained by CAM [65] is prone to focus on small discriminative parts.

CAM is able to make the network attend to more object parts and produce more uniform response distributions on objects.

Effectiveness of Regularization. One interesting aspect we find is that adding regularization could enhance the effectiveness of the refinement step. In Table 2.1, compared to Mixup $\mathcal{L}_{cls}$, adding either $\mathcal{L}_{ent}$ (3rd row) or $\mathcal{L}_{con}$ (4th row) improves the CAM IoU by 0.2% and 0.6% respectively. Nevertheless, with the refinement, the corresponding improvements in IoU is 1.1% and 1.2%, which are larger than the ones before refining the response. This is because our regularization enforces a more uniform response on objects, which greatly facilitates the refinement step (*e.g.*, via region expanding). In addition, we illustrate one example in Figure 2.4, where the IoU difference of initial response is relatively small, but the resultant refined map could differ significantly.

Parameter Sensitivity. In this chapter, we mainly study three parameters in our Mixup-CAM framework, i.e., α for mixup regularization and $\{\lambda_{ent}, \lambda_{con}\}$ in uncertainty regularization. In Figure 2.5(a), when increasing the α value, the *Beta* distribution would become more uniform, which encourages a more uniform λ_{mix} in (2.2) and results in mixed images that are more challenging to optimize. Nevertheless, the IoUs of Mixup $\mathcal{L}_{cls}$ under various α are consistently better than the CAM baseline. For regularization terms, we fix $\lambda_{con} = 2e-4$ and adjust λ_{ent} in Figure 2.5(b), while fixing $\lambda_{ent} = 0.2$ and

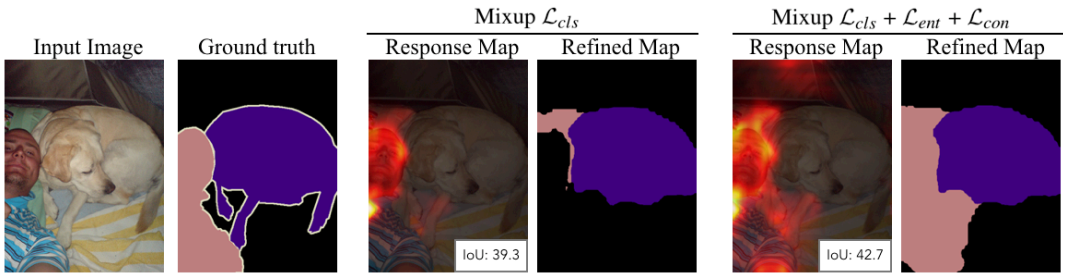

Figure 2.4: Enhancement on refinement. Our regularization enforces a more uniform response on objects, which can facilitate the refinement step. The examples illustrate that the IoU difference of the resultant refined map is significantly larger than the one of initial response.

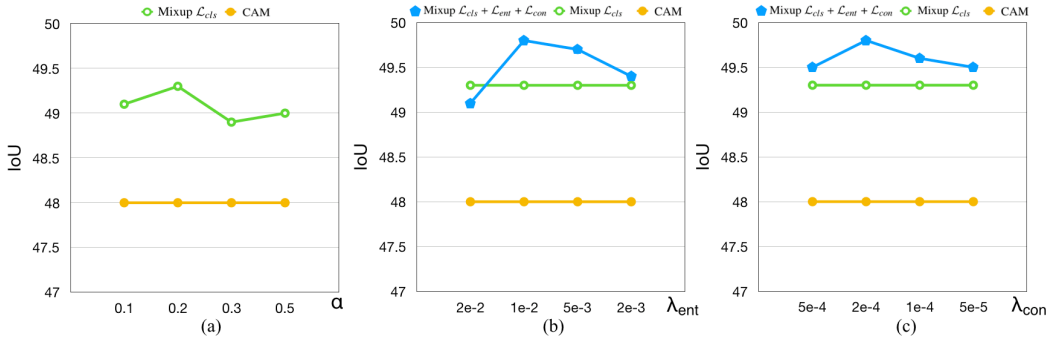

Figure 2.5: Sensitivity analysis for parameters. (a) α for mixup augmentation; (b) λ_{ent} and (c) λ_{con} for uncertainty regularization.

change λ_{con} in Figure 2.5(c). Both figures show that these two parameters are robust to the performance over a wide range of values.

2.4.3 Semantic Segmentation Performance

After generating the pseudo ground truths using the refined response map, we use them to train the semantic segmentation network. First, we compare our method with state-of-the-art algorithms using the ResNet-101 architecture or other similarly powerful ones in Table 2.2. Note that, while our method focuses on improving the initial responses on the object, most methods aim to improve the refinement step or segmentation network

Table 2.2: Comparison of WSSS methods using image-level labels on the PASCAL VOC 2012 validation set. ✓ indicates the methods that focus on improving the initial response. The result of † on AffinityNet is re-produced by training the same ResNet-101 as our pipeline.

Method	Backbone	Init. Resp.	IoU on Val
MCOF $_{\text{CVPR'18}}$ [55]	ResNet-101		60.3
DCSP $_{\text{BMVC'17}}$ [5]	ResNet-101		60.8
DSRG $_{\text{CVPR'18}}$ [26]	ResNet-101		61.4
AffinityNet $_{\text{CVPR'18}}$ [1]	Wide ResNet-38		61.7
AffinityNet† $_{\text{CVPR'18}}$ [1]	ResNet-101		61.9
SeeNet $_{\text{NIPS'18}}$ [25]	ResNet-101	✓	63.1
Zeng *et al* $_{\text{ICCV'19}}$ [61]	DenseNet-169		63.3
BDSSW $_{\text{ECCV'18}}$ [18]	ResNet-101		63.6
OAA $_{\text{ICCV'19}}$ [28]	ResNet-101	✓	63.9
CIAN $_{\text{CVPR'19}}$ [17]	ResNet-101		64.1
FickleNet $_{\text{CVPR'19}}$ [33]	ResNet-101	✓	64.9
SSDD $_{\text{ICCV'19}}$ [47]	Wide ResNet-38		64.9
Ours	ResNet101	✓	**65.6**

training. In Table 2.3, we further present detailed performance for each category. We show two groups of results without (top rows) or with (bottom rows) applying CRF [32] to refine final segmentation outputs. Compared to the recent FickleNet [33] approach that also tries to improve the initial response map, our proposed Mixup-CAM shows favorable performance in final semantic segmentation results.

2.4.4 Qualitative Comparisons

Figure 2.6 presents more initial response maps generated by the CAM [65] method and ours. A number of qualitative examples of our final semantic segmentation results are presented in Figure 2.7. In addition, we show some failure segmentation cases in Figure 2.8. There are three main issues that would affect the quality of segments: 1) the incompleteness on detailed parts, 2) the ambiguity on object boundaries, and 3) the noise on background. The first two issues are the common problem of the WSSS task. The third issue could be raised by the increased uncertainty from mixup augmentations,

Table 2.3: Semantic segmentation performance on the PASCAL VOC 2012 validation set. The bottom group contains results with CRF refinement, while the top group is without CRF. The best three results are in red, green and blue, respectively.

Method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mIoU
AffinityNet [1]	88.2	68.2	30.6	81.1	49.6	61.0	77.8	66.1	75.1	29.0	66.0	40.2	80.4	62.0	70.4	73.7	42.5	70.7	42.6	68.1	51.6	61.7
Ours (w/o CRF)	87.6	54.5	30.7	73.0	46.5	72.0	86.5	74.8	87.6	31.3	80.8	50.3	82.6	74.5	67.2	68.7	39.6	79.2	44.8	64.9	51.1	64.2
MCOF [55]	87.0	78.4	29.4	68.0	44.0	67.3	80.3	74.1	82.2	21.1	70.7	28.2	73.2	71.5	67.2	53.0	47.7	74.5	32.4	71.0	45.8	60.3
Zeng et al. [61]	90.0	77.4	37.5	80.7	61.6	67.9	81.8	69.0	83.7	13.6	79.4	23.3	78.0	75.3	71.4	68.1	35.2	78.2	32.5	75.5	48.0	63.3
FickleNet [33]	89.5	76.6	32.6	74.6	51.5	71.1	83.4	74.4	83.6	24.1	73.4	47.4	78.2	74.0	68.8	73.2	47.8	79.9	37.0	57.3	64.6	64.9
SSDD [47]	89.0	62.5	28.9	83.7	52.9	59.5	77.6	73.7	87.0	34.0	83.7	47.6	84.1	77.0	73.9	69.6	29.8	84.0	43.2	68.0	53.4	64.9
Ours (w/ CRF)	88.4	57.0	31.2	75.2	47.8	72.4	87.2	76.0	89.2	32.7	83.1	51.1	85.4	77.3	68.4	70.2	40.0	81.5	46.2	65.4	51.8	65.6

such that the expanded initial response could attend to the non-object region. Although there are some failure examples, our approach generally produces high-quality semantic segmentation results.

2.5 Summary

In this chapter, we propose the Mixup-CAM framework to improve the localization of object response maps, as an initial step towards weakly-supervised semantic segmentation task using image-level labels. To this end, we propose to integrate the mixup data augmentation strategy for calibrating the uncertainty in network prediction. Furthermore, we introduce another two regularization terms as the interplay with the mixup scheme, thereby producing more complete and uniform response maps. In experimental results, we provide comprehensive analysis of each component in the proposed method and show that our approach achieves state-of-the-art performance against existing algorithms.

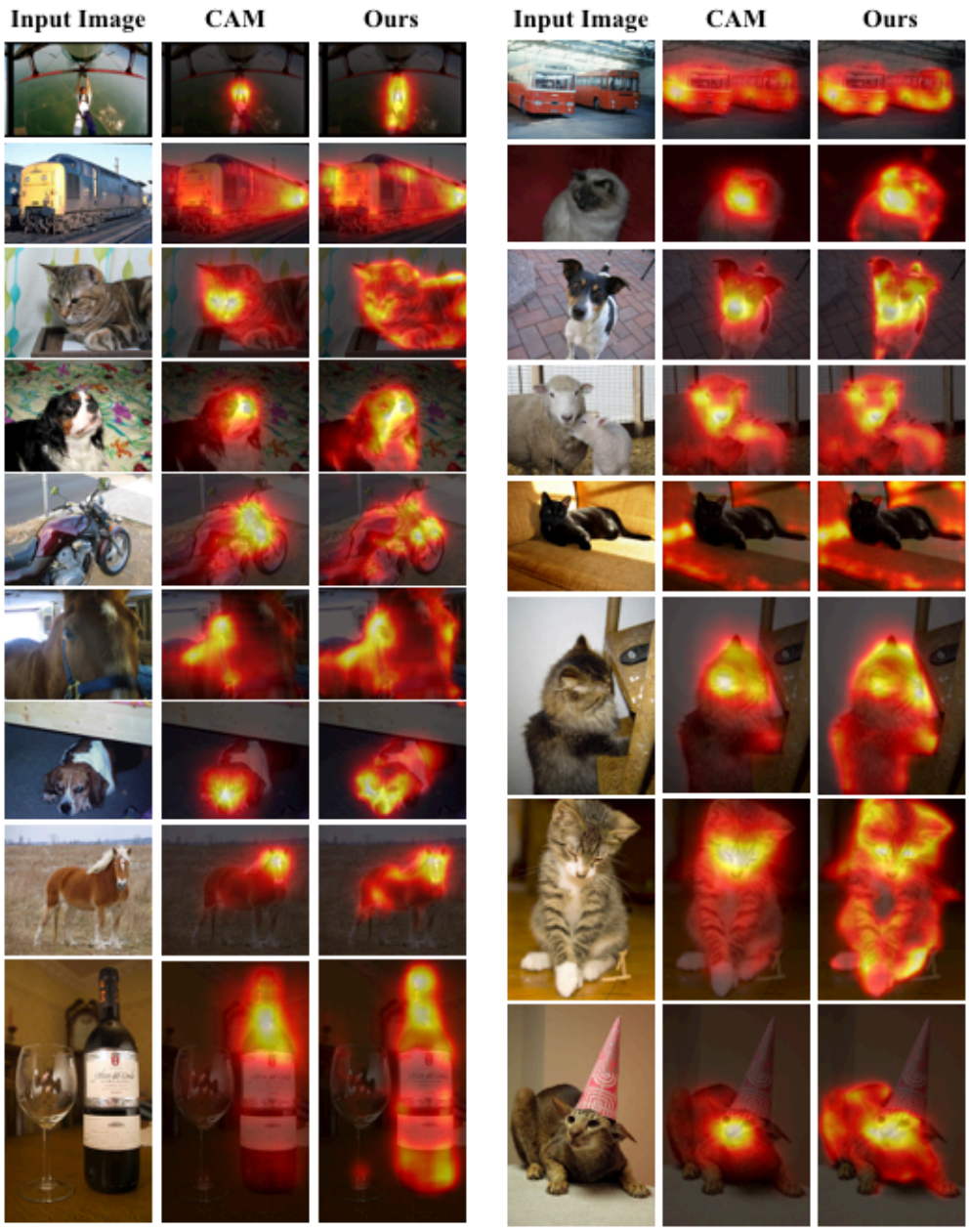

Figure 2.6: Qualitative comparison of the initial response map with [65] on the PASCAL VOC 2012 val images.

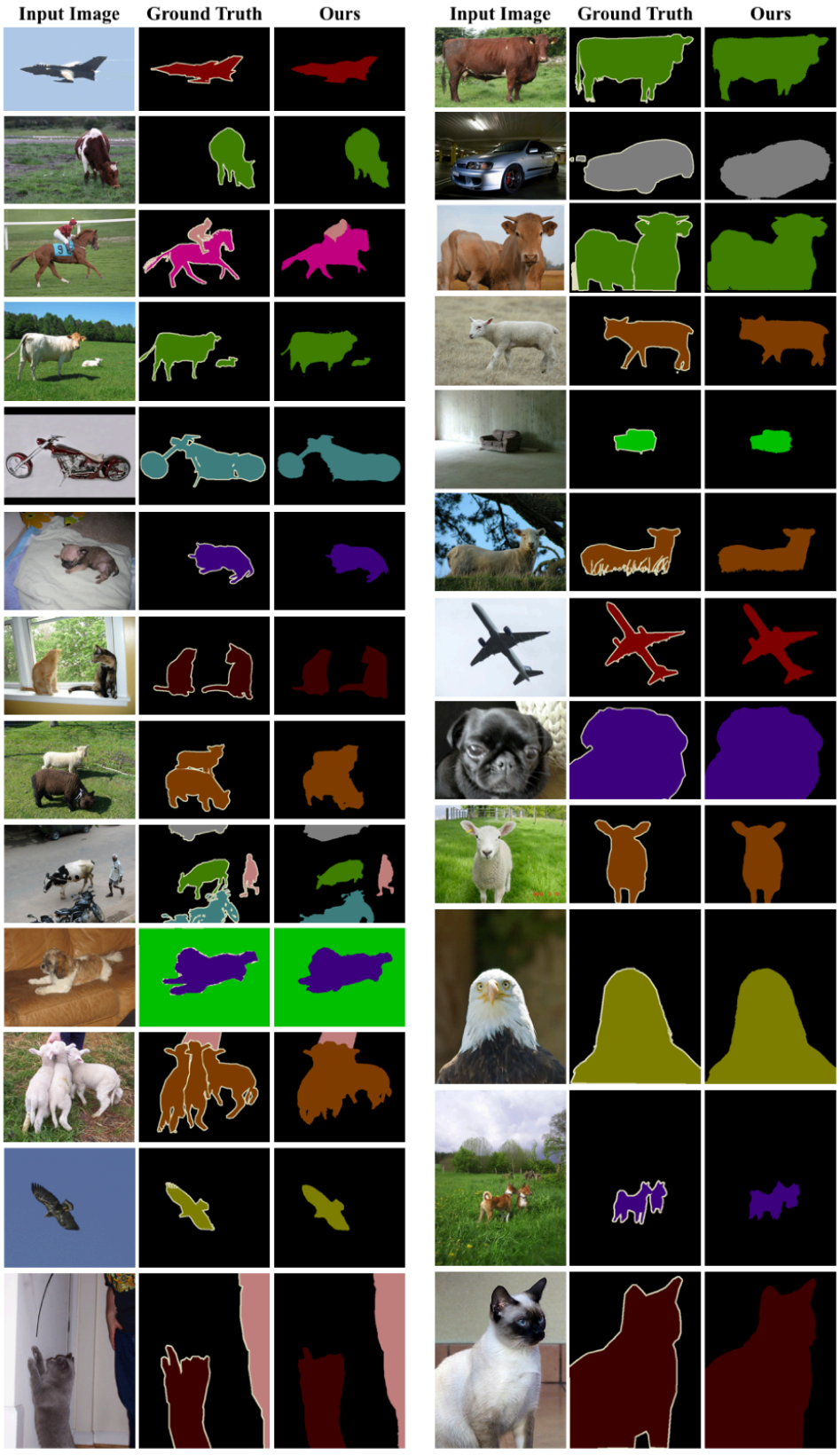

Figure 2.7: Semantic segmentation results on the PASCAL VOC 2012 val images.

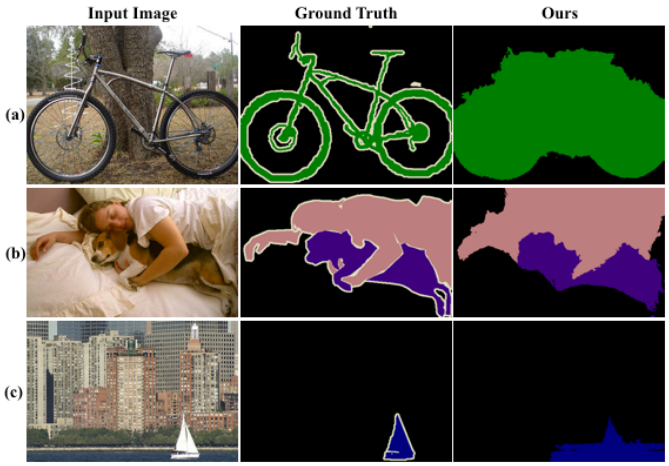

Figure 2.8: Failure semantic segmentation examples. (a) Missing detailed parts. Details of the bicycle are missing in the segment. (b) The ambiguity on object boundary. There are errors on the boundary region between two objects. (c) The background noise.

Chapter 3

Weakly-Supervised Semantic Segmentation via Sub-category Exploration

3.1 Introduction

The goal of semantic segmentation is to assign a semantic category to each pixel in the image. It has been one of the most important tasks in computer vision that enjoys a wide range of applications such as image editing and scene understanding. Recently, deep convolutional neural network (CNN) based methods [24, 6, 59] have been developed for semantic segmentation and achieved significant progress. However, such approaches rely on learning supervised models that require pixel-wise annotations, which take extensive effort and time. To reduce the effort in annotating pixel-wise ground truth labels, numerous weakly-supervised methods are proposed using various types of labels such as image-level [1, 31, 41, 45], video-level [8, 64, 51], bounding box [40, 12, 29], point-level [2], and scribble-based [36, 54] labels. In this work, we focus on using image-level labels which can be obtained effortlessly, yet are a more challenging case under the weakly-supervised setting.

Existing algorithms mainly consist of three sequential steps to perform weakly-supervised training on the image-level label: 1) predict an initial category-wise response

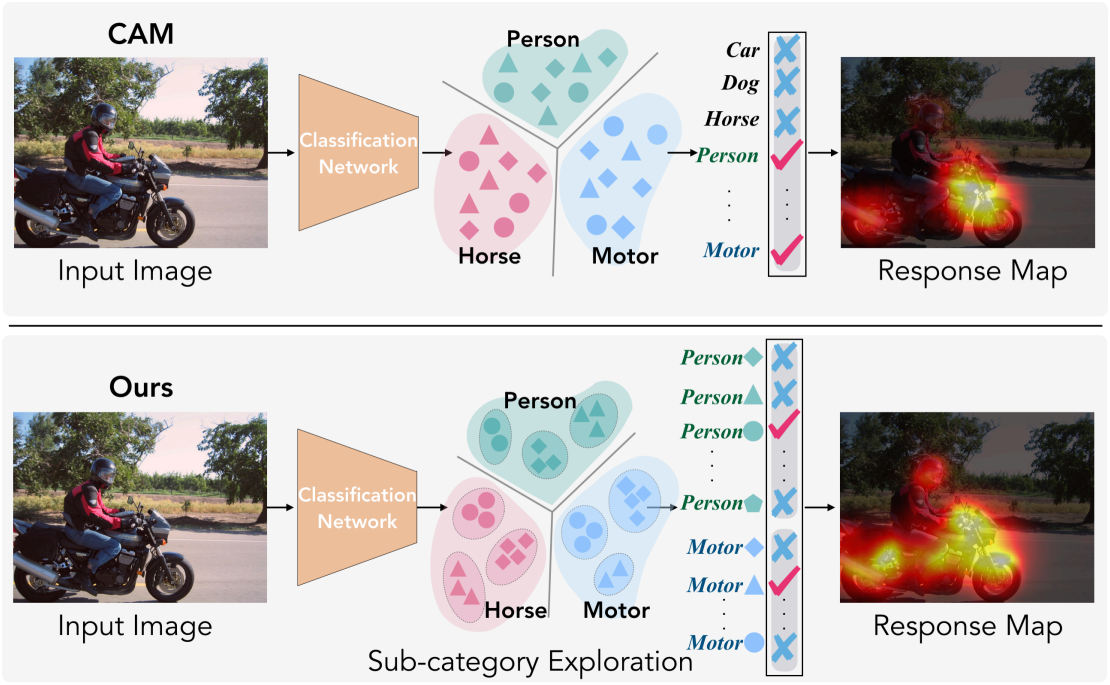

Figure 3.1: Existing weakly-supervised semantic segmentation methods based on image-level supervisions usually apply the class activation map (CAM) to obtain the response map as the initial prediction. However, this response map can only highlight the discriminative parts of the object (top). We propose a self-supervised task via sub-category exploration to enforce the classification network learn better response maps (bottom).

map to localize the object, 2) refine the initial response as the pseudo ground truth, and 3) train the segmentation network based on pseudo labels. Although promising results have been achieved by recent methods [1, 26, 55, 57], most of them focus on improving the second and the third steps. Therefore, these approaches may suffer from inaccurate predictions generated in the first step, i.e., initial response. Here, we aim to improve the performance of initial predictions which will benefit succeeding steps.

In order to predict the initial response map for each category, numerous approaches based on the class activation map (CAM) model [65] have been developed. Essentially, these methods train a classification network and use its learned weights in the classifier as the cues to compute weighted sums of feature maps, which can be treated as the response map. However, such response maps may only focus on a portion of the object,

instead of localizing the entire object (see top of Figure 3.1). One explanation is that the objective of the classifier does not need to "see" the entire object for optimizing the loss function. This impairs the classifier's ability to locate the objects.

At the core of our technique is to impose a more challenging task to the network for learning better representations, while not jeopardizing the original objective. To this end, we propose a simple yet effective method by introducing a self-supervised task that discovers sub-categories in an unsupervised manner, as illustrated at the bottom of Figure 3.1. Specifically, our task consists of two steps: 1) perform clustering on image features extracted from the classification network for each annotated parent class (e.g., 20 parent classes on the PASCAL VOC 2012 dataset [16]), and 2) use the clustering assignment for each image as the pseudo label to optimize the sub-category objective.

On one hand, the parent classifier establishes a feature space through supervised training as the guidance for unsupervised sub-category clustering. On the other hand, the sub-category objective provides additional gradients to enhance feature representations and leverage the sub-space of the original feature space to obtain better results. As such, the classification model takes a more challenging task and is not limited to the easier objective of learning only the parent classifier. Moreover, to ensure better convergence in practice, we iteratively alter the two steps of feature clustering and pseudo training the sub-category objective.

We conduct extensive experiments on the PASCAL VOC 2012 dataset [16] to demonstrate the effectiveness of our method, with regard to generating better initial response maps to localize objects. As a result, our approach leads to favorable performance for the final semantic segmentation results against state-of-the-art weakly-supervised approaches. Furthermore, we provide extensive ablation studies and analysis to validate the robustness of our method. Interestingly, we notice that the network is able to differentiate sub-categories with respect to their object size/type, context, and coexistence with other categories. The main contributions of this work are summarized as follows:

- We propose a simple yet effective method via a self-supervised task to enhance feature representations in the classification network. This improves the initial class activation maps for weakly-supervised semantic segmentation as well.

- We explore the idea of sub-category discovery via iteratively performing unsu-

pervised clustering and pseudo training on the sub-category objective in a self-supervised fashion.

- We present extensive study and analysis to show the efficacy of the proposed method, which significantly improves the quality of initial response maps and leads to better semantic segmentation results.

3.2 Related Work

Within the context of this work, we discuss methods for weakly-supervised semantic segmentation (WSSS) using image-level labels, including approaches that focus on initial prediction and refinement for generating pseudo ground truths. In addition, algorithms that are relevant to unsupervised representation learning are discussed in this section.

Initial Prediction for WSSS. Initial cues are essential for segmentation task since it can provide reliable priors to generate segmentation maps. The class activation map [65] is a widely used technique for localizing the object. It can highlight class-specific regions that serve as the initial cues. However, since the CAM model is trained by a classification task, it tends to activate to the small discriminative part of the object, leading to incomplete initial masks.

Several methods have been developed to alleviate this problem. Numerous approaches [48, 56] deliberately hide or erase the region of an object, forcing models to seek more diverse parts. However, those methods either hide fixed-size patches randomly or require repetitive model training and response aggregation steps. A number of variants [63, 34] have been proposed to extend the initial response via an adversarial erasing strategy in an end-to-end training manner, yet such strategies may gradually expand their attention to non-object regions, leading to inaccurate attention maps. Recently, the SeeNet approach [25] applies self-erasing strategies to encourage networks to use both object and background cues, which prevent the attention from including more background regions. Instead of using the erasing scheme, the FickleNet method [33] introduces stochastic feature selection to obtain diverse combinations of locations

on feature maps. By aggregating the localization maps, they acquire the initial cue that contains a larger region of the object.

Different from the methods that mitigate the problem by discovering complementary regions via iterative erasing steps or consolidating attention maps, our proposed approach aims at forcing the network to learn on a more challenging task via self-supervised sub-category exploration, thereby enhancing feature representations and improving the response map.

Response Refinement for WSSS. Numerous approaches [1, 17, 18, 26, 31, 55, 57] have been proposed to refine the initial cue via expanding the region of attention map. The SEC method [31] proposes a loss function that constrains both global weighted rank pooling and low-level boundary to expand the localization map. To improve the network training, the MCOF scheme [55] uses a bottom-up and top-down framework which alternatively expands object regions and optimizes the segmentation network, while the MDC method [57] expands the seeds by employing multiple branches of convolutional layers with different dilation rates. Moreover, the DSRG approach [26] refines initial localization maps by applying a seeded region growing method during the training of the segmentation network. Other approaches have been developed via affinity learning. For instance, the AffinityNet [1] considers pixel-wise affinity to propagate local responses to nearby areas, while [17, 18] explore cross-image relationships to obtain complementary information that can infer the predictions.

Nevertheless, initial seeds are still obtained from the CAM method. If these seeds only come from the discriminative parts of objects, it is difficult to expand regions into non-discriminative parts. Moreover, if the initial prediction produces wrong attention regions, applying the refinement step would cover even more inaccurate regions. In this thesis, we focus on improving the initial prediction, which leads to more accurate object localization and benefits the refinement step.

Unsupervised Representation Learning. Unsupervised learning has been widely studied in the computer vision community. One advantage is to learn better representations of images and apply learned features on any specific domain or dataset where annotations are not always available. Self-supervised learning [13] utilizes a pretext task to

replace the labels annotated by humans with "pseudo-labels" directly computed from the raw input data. A number of methods [39, 42, 43] have been developed but require expert knowledge to carefully design a pretext task that may lead to good transferable features. To reduce the domain knowledge requirement, Coates and Ng [9] validate that feature-learning systems with K-means can be a scalable unsupervised learning module that can train a model of the unlabeled data for extracting meaningful features. Furthermore, a recent approach [4] employs a clustering framework to extract useful visual features by alternating between clustering the image descriptors and updating the weights of the CNN by predicting the cluster assignments, in order to learn deep representations specific to domains where annotations are scarce. In this work, we propose to learn a self-supervised method that explores the sub-category in the classification network, i.e., using unsupervised signals to enhance feature representations while improving initial response maps for weakly-supervised semantic segmentation.

3.3 Proposed Algorithm

In this section, we describe our framework for weakly-supervised semantic segmentation, including details of how we explore sub-categories to improve initial response maps and generate final semantic segmentation results.

3.3.1 Algorithm Overview

To obtain the initial response, we follow the common practice of training a classification network and utilize the CAM method [65] to obtain our baseline model. The CAM method typically only activates on discriminative object parts, which are not sufficient for the image classification task. To address this issue, we propose to integrate a more challenging task into the objective: self-supervised sub-category discovery, in order to force the network to learn from more object parts.

Firstly, for each annotated parent class, we determine K sub-categories by applying K-means clustering on image features. With the clustering results, we then assign each image a pseudo label, which is identified as the index of the sub-category. Finally, we construct a sub-category objective to jointly train the classification network. By

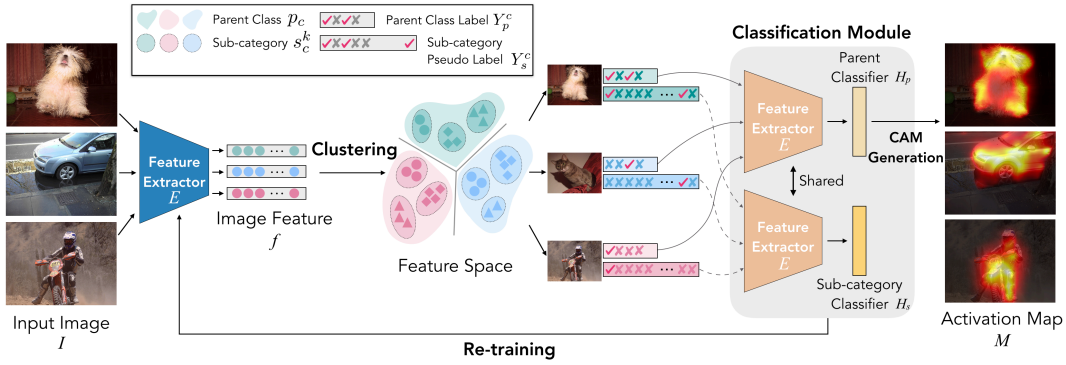

Figure 3.2: Proposed framework for generating the class activation map. Given input images I, we first feed them into a feature extractor E to obtain their features f. Then, we adopt unsupervised clustering on f and obtain sub-category pseudo labels Y_s for each image. Next, we train the classification network to jointly optimize the parent classifier H_p with ground truth labels Y_p for parent classes and the sub-category classifier H_s using the sub-category pseudo labels obtained in the clustering stage. By iteratively performing unsupervised clustering on image features and pseudo training the classification module, we use the jointly optimized classification network to produce the final activation map M.

iteratively updating the feature extractor, two classifiers, and sub-category pseudo labels, the enhanced features representations lead to better classification, and thereby gradually produce response maps that attain to more complete regions of the objects. The overall process is illustrated in Figure 3.2. Then, we use the method in [1] to expand response maps, which are used as pseudo ground truths to train the segmentation network. Also note that, our method focuses on the initial prediction, so it is not limited to certain region expansion or segmentation training methods.

Preliminaries: Initial Response via CAM. We adopt the CAM to generate the initial response using a typical classification network, whose architecture consists of convolutional layers as the feature extractor E, followed by global average pooling (GAP) and one fully-connected layer H_p as the output classifier. Given an input image I, the network is trained with image-level labels Y_p using a multi-label classification loss $\mathcal{L}_p$, following [65]. After training, the activation map M for each category c can be obtained

via directly applying classifier H_p on the feature maps $f = E(I)$:

$$M^c(x, y) = \theta_p^{c\top} f(x, y), \tag{3.1}$$

where θ_p^c is the classifier weight for the category c, and $f(x, y)$ is the feature at pixel (x, y). The response map is further normalized by the maximum value in M^c.

3.3.2 Sub-category Exploration

The activation map for each image using (3.1) provides typically highlights only the discriminative object parts. However, from the perspective of a classifier, discovering the most discriminative part of the object is already sufficient for optimizing the loss function $\mathcal{L}_p$ in classification. As the learning objective is based on the classification scores, it is inevitable for the CAM model to generate incomplete attention maps. To address this issue, we integrate a self-supervised scheme to enhance feature representations f while improving the response maps via exploring the sub-category information, in which f appears to be an important cue to compute the activation map via (3.1).

Sub-Category Objective. To assign a more challenging problem to the classification model, we introduce a task to discover sub-categories in an unsupervised manner. For each parent class p_c, we define K sub-categories s_c^k, where $k = \{1, 2, , K\}$. For each image I with the parent label Y_p^c in $\{0, 1\}^c$, the corresponding sub-category label for the category c is denoted as $Y_s^{c,k}$ in $\{0, 1\}^k$. We also note that, if the label of one parent class does not exist (i.e., $Y_p^c = 0$), the labels of all sub-categories would be also 0, i.e., $Y_s^{c,k} = 0, k = \{1, 2, ..., K\}$. Our objective is to learn a sub-category classifier H_s parameterized with θ_s, while sharing the same feature extractor E with H_p. Similar to the parent classification loss $\mathcal{L}_p$, we adopt the standard multi-label classification loss $\mathcal{L}_s$ with a larger and fine-grained label space Y_s.

Sub-category Discovery. As there is no ground truth label for sub-category to directly optimize the above sub-category objective $\mathcal{L}_s$, we generate pseudo labels via unsupervised clustering. Specifically, we perform clustering for each parent class on image features extracted from the feature extractor E. The clustering objective for each class

c can be written as:

$$\min_{D\in\mathbb{R}^{d\times k}} \frac{1}{N^c}\sum_{i=1}^{N^c}\min_{Y_s^c}||f - TY_s^c||_2^2, \quad \text{s.t.}, Y_s^{c\top}1_k = 1, \tag{3.2}$$

where T is a $D \times K$ centroid matrix, N^c is the number of images containing the class c, and $f = E(I) \in \mathbb{R}^D$ is the extracted feature. We use the clustering assignment Y_s^c for each image as the sub-category pseudo label to optimize $\mathcal{L}_s$.

Joint Training. After obtaining sub-category pseudo labels Y_s from the above clustering process, we jointly optimize the feature representations $f = E(I)$ and two classifiers, i.e., H_p and H_s:

$$\min_{\theta_p,\theta_s} \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_p(H_p(f_i), Y_p) + \lambda\mathcal{L}_s(H_s(f_i), Y_s), \tag{3.3}$$

where N is the total number of images and λ is weight to balance two loss functions. With this method, the parent classification learns a feature space through supervised training via $\mathcal{L}_p$, while the sub-category objective $\mathcal{L}_s$ explores the feature sub-space and provides additional gradients to enhance feature representations f, which is used to compute CAM via (3.1).

Iterative Optimization. The proposed unsupervised clustering scheme in (3.2) relies on the feature f to discover sub-category pseudo labels. As such, the learned features via only the objective $\mathcal{L}_p$ could be less discriminative for the clustering purpose. To mitigate this issue, we adopt an iterative training method by alternatively updating (3.2) and (3.3). Therefore, features f are first enhanced through the sub-category objective, and in turn facilitate the clustering process to generate better pseudo ground truths, which are then used to learn better feature representations in network training. The overall optimization for generating final class activation maps is summarized in Algorithm 1.

3.3.3 Implementation Details

In this section, we describe the implementation details of the proposed framework and the following procedures to produce final semantic segmentation results.

Algorithm 1 Learning Sub-category Discovery for CAM

Input: Image I; Parent Label Y_p; Category Number C;
Sub-category Number K
Output: Class Activation Map M^c
Model: Feature extractor E; Parent Classifier $(H_p; \theta_p)$;
Sub-category Classifier $(H_s; \theta_s)$

Optimize $\{E, H_p\}$ with Y_p via $\mathcal{L}_p$
while Training **do**
  Extract features via $f = E(I)$
  for $c \leftarrow 1$ to C **do**
    Generate pseudo labels Y_s^c with f via (3.2)
  Optimize $\{E, H_p, H_s\}$ with $\{Y_p, Y_s\}$ via (3.3)
Compute M^c via (3.1)

Classification Network. In this work, the ResNet-38 architecture [58] is used for the CAM model, and the training procedure is similar to that in [1]. The network consists of 38 convolution layers with wide channels, followed by a 3×3 convolution layer with 512 channels for better adaptation to the classification task, a global average pooling layer for feature aggregation, and two fully-connected layers for image and sub-category classification, respectively. The model is pre-trained on the ImageNet [14] and is then finetuned on the PASCAL VOC 2012 dataset. We use typical techniques such as horizontal flip, random cropping, and color jittering operations to augment the training data set. We also randomly scale the input images to impose scale invariance in the network.

We implement the proposed framework with PyTorch and train on a single Titan X GPU with 12 GB memory. To train the classification network, we use the Adam optimizer [30] with initial learning rate of 1e-3 and the weight decay of 5e-4. In practice, we use $\lambda = 5$ and $K = 10$ in all the experiments unless specified otherwise. For iterative training, we empirically find that the model converges after training for 3 rounds. In the experimental section, we show studies for the choice of K and iterative training results.

Semantic Segmentation Generation. Based on the response map generated by our method as in Algorithm 1, we adopt the random walk method via affinity [1] to refine the map as pixel-wise pseudo ground truths for semantic segmentation. In addition, as a common practice, we use dense conditional random fields (CRF) [32] to further refine the response to obtain better object boundaries. To train the segmentation network, we utilize the Deeplab-v2 framework [6] with the ResNet-101 architecture [23] as the backbone model.

3.4 Experimental Results

In this section, we first present the main results and analysis of the initial response generated by our method. Second, we show the final semantic segmentation performance on the PASCAL VOC dataset [16] against the state-of-the-art approaches.

3.4.1 Evaluated Dataset and Metric

We evaluate the proposed approach on the PASCAL VOC 2012 semantic segmentation benchmark [16] which contains 21 categories, including one background class. Each image contains one or multiple object classes. Following previous weakly-supervised semantic segmentation methods, we use the augmented 10,528 training images present in [21] along with their image-level labels to train the network. To evaluate the training set, we use the set without augmentation which has 1,464 examples. We adopt 1,449 images in the validation set and 1,456 images in the test set to compare our results with other methods. For all experiments, the mean Intersection-over-Union (mIoU) ratio is used as the evaluation metric. The results for the test set are obtained from the official PASCAL VOC evaluation website.

3.4.2 Improvement on Initial Response

In Table 3.1, we show the mean IoU of the segments computed using the CAM on both the training and validation sets. We present results after applying the refinement step to the activation map, i.e., CAM + random walk (CAM + RW). Table 3.1 shows

Table 3.1: Performance comparison in mIoU (%) for evaluating activation maps on the PASCAL VOC training and validation sets.

	Training Set		Validation Set	
Method	CAM	CAM+RW	CAM	CAM+RW
AffinityNet [1]	48.0	58.1	46.8	57.0
Ours	50.9	63.4	49.6	61.2

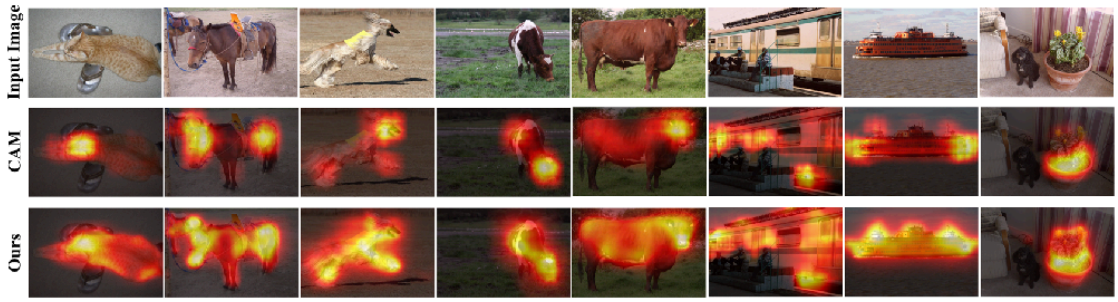

Figure 3.3: Sample results of initial responses. Our method often generates the response map that covers larger region of the object (i.e., attention on the body of the animal), while the response map produced by CAM [65] tends to highlight small discriminative parts.

that our approach significantly improves the IoU over AffinityNet [1] by almost 3% using CAM and more than 4% for CAM+RW. The improved initial response maps facilitate the downstream task in generating pixel-wise pseudo ground truths for training the semantic segmentation model.

In Figure 3.3, we show comparisons of generated CAMs by the conventional classification loss $\mathcal{L}_p$ [65] and the proposed method via sub-category discovery summarized in Algorithm 1. Visual results show that our method is able to localize more complete object regions, while the original CAM only focuses on discriminative object parts. We also note that this is essentially critical for the refinement stage that takes the response map as the input.

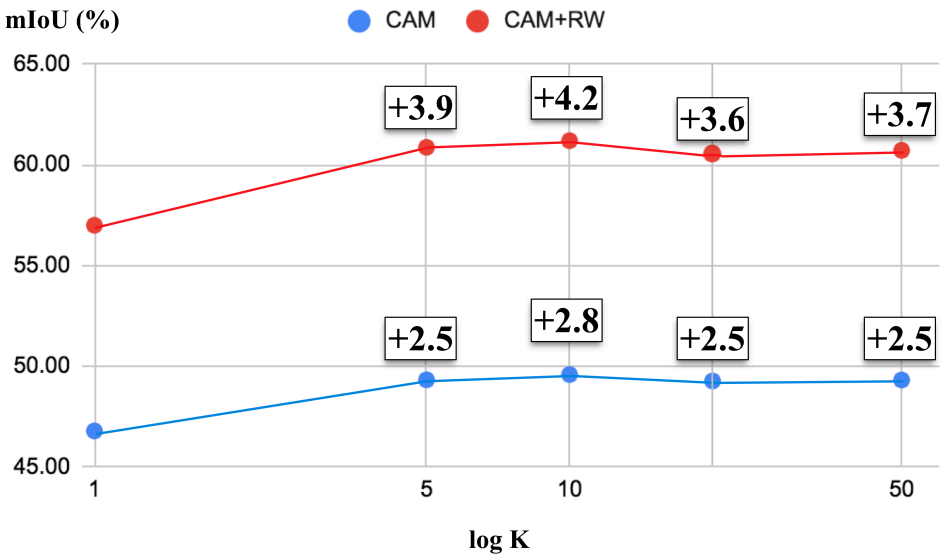

Figure 3.4: Ablation study for K. We show that the proposed method performs robustly with respect to K and is consistently better than the original CAM that did not apply clustering to discover sub-categories. We mark the value of mIoU of the original CAM at $K = 1$ and the improved mIoUs are presented.

3.4.3 Ablation Study and Analysis

To demonstrate how our method helps improve feature representations and allow the network to pay more attention to other object parts via exploiting the sub-category information, we present extensive analysis in this section. Here, all the experimental results are based on the PASCAL VOC validation set.

Effect of Sub-Category Number K. We first study how the sub-category number K affects the performance of the proposed method. In Figure 3.4, we use $K = \{5, 10, 20, 50\}$, and show that the proposed method performs robustly with respect to K (within a wide range) and consistently better than the original CAM method (i.e., $K = 1$). The results also validate the necessity and importance of using more sub-categories (i.e., $K > 1$) to generate better response maps. Considering the efficiency and accuracy, we use $K = 10$ for each parent class in all the experiments. As a future work, it is of great interest to develop an adaptive method to determine the sub-category number [46], which can reduce the redundant sub-categories and make the approach more efficient.

Table 3.2: Segmentation quality of the initial response at different rounds of training on the PASCAL VOC 2012 validation set. We show there is a gradual improvement on both mIoU and F-Score metrics.

Round	mIoU (%) ↑	F-Score ↑
#0 (CAM)	46.8	65.1
#1	48.0	65.6
#2	48.7	66.6
#3	**49.6**	**67.0**

Iterative Improvement. To demonstrate the effectiveness of our iterative training process, we show the gradual improvement on the segment quality in Table 3.2. We present the results of mIoU and F-Score that accounts for both the recall and precision measurements, in which they are important cues to validate whether the activation map is able to cover object parts. Compared to the results in round #0, which is the original CAM, our method gradually improves both metrics as training more rounds.

Clustering Results. Since the ground truth labels are not available for sub-categories, we present visualizations of the clustering results in Figure 3.5 to measure the quality, in which each parent class shows 3 example clusters. Our method is able to cluster objects based on their size (*Aeroplane, Bird, Cow*), context (*Aeroplane, Bird, Person*), type (*Boat, Bird*), pose (*Cow*), and interaction with other categories (*Person*). For instance, persons with different categories, e.g., horse, motobike, and boat, are clustered into different groups. This visually validates that our learned feature representations are enhanced via the sub-category objective in an unsupervised manner.

Weight Visualization. In order to understand how our learning mechanism improves the clustering quality, we visualize the distribution of the classifier weights, i.e., θ_p and θ_s, via t-SNE [52]. As such, we are able to find the relationship between the parent classifier H_p and the sub-category module H_s. Figure 3.6 shows the visualization of

Figure 3.5: Clustering results of the last round model (#3). We show 3 clusters for each parent class and demonstrate that our learned features are able to cluster objects based on their size (*Aeroplane, Bird, Cow*), context (*Aeroplane, Bird, Person*), type (*Boat, Bird*), pose (*Cow*), and interaction with other categories (*Person*).

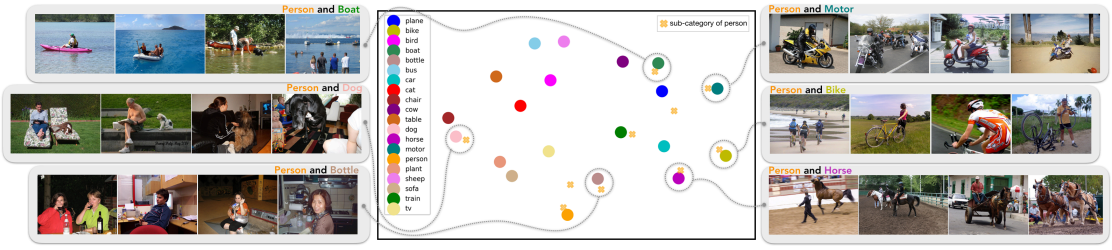

Figure 3.6: Visualizations of weights based on the t-SNE method that illustrates the relationships on semantic-level between parent classifier and the person sub-category classifier. We show that one person sub-category is usually close to one parent class, as they often co-appear in the same image, as shown in example images on two sides.

weights, in which we take the sub-categories of person (denoted as yellow cross symbols) as the example, since the person category has more interactions with other parent classes (denoted as solid circles). It illustrates that one person sub-category is often close to one parent class, e.g., sub-category *person* and parent class *bike*, which makes sense as those two categories usually co-appear in the same image (see example images in Figure 3.6 on two sides).

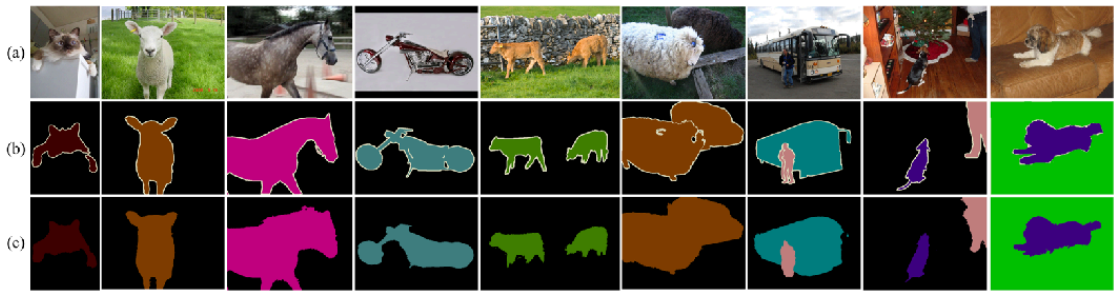

Figure 3.7: Qualitative results on the PASCAL VOC 2012 validation set. (a) Input images. (b) Ground truth. (c) Our results.

Table 3.3: Semantic segmentation performance on the PASCAL VOC 2012 validation set. Bottom group contains results with CRF refinement, while the top group is without CRF. Note that 11/20 classes obtain improvements using our approach w/ CRF. The best three results are in red, green and blue, respectively.

Method	bkg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mIoU
AffinityNet [1]	88.2	68.2	30.6	81.1	49.6	61.0	77.8	66.1	75.1	29.0	66.0	40.2	80.4	62.0	70.4	73.7	42.5	70.7	42.6	68.1	51.6	61.7
Ours (w/o CRF)	88.1	49.6	30.0	79.8	51.9	74.6	87.7	73.7	85.1	31.0	77.6	53.2	80.3	76.3	69.6	69.7	40.7	75.7	42.6	66.1	58.2	64.8
MCOF [55]	87.0	78.4	29.4	68.0	44.0	67.3	80.3	74.1	82.2	21.1	70.7	28.2	73.2	71.5	67.2	53.0	47.7	74.5	32.4	71.0	45.8	60.3
Zeng et al. [61]	90.0	77.4	37.5	80.7	61.6	67.9	81.8	69.0	83.7	13.6	79.4	23.3	78.0	75.3	71.4	68.1	35.2	78.2	32.5	75.5	48.0	63.3
FickleNet [33]	89.5	76.6	32.6	74.6	51.5	71.1	83.4	74.4	83.6	24.1	73.4	47.4	78.2	74.0	68.8	73.2	47.8	79.9	37.0	57.3	64.6	64.9
Ours (w/ CRF)	88.8	51.6	30.3	82.9	53.0	75.8	88.6	74.8	86.6	32.4	79.9	53.8	82.3	78.5	70.4	71.2	40.2	78.3	42.9	66.8	58.8	66.1

3.4.4 Semantic Segmentation Performance

After generating the pseudo ground truths as the results in Table 3.1 (i.e., CAM + RW), we use them to train the semantic segmentation network. We first compare our method with recent work using the ResNet-101 backbone or other similarly powerful ones in Table 3.4. On both validation and testing sets, the proposed algorithm performs favorably against the state-of-the-art approaches. We also note that, most methods focus on improving the refinement stage or network training, while ours improves the initial step to generate better object response maps.

Table 3.4: Comparison of weakly-supervised semantic segmentation methods on the PASCAL VOC 2012 val and test sets. In addition, we present methods that aim to improve the initial response with ✓ in the "Init. Res." column.

Method	Backbone	Init. Res.	Val	Test
MCOF $_{\text{CVPR'18}}$ [55]	ResNet-101		60.3	61.2
DCSP $_{\text{BMVC'17}}$ [5]	ResNet-101		60.8	61.9
DSRG $_{\text{CVPR'18}}$ [26]	ResNet-101		61.4	63.2
AffinityNet $_{\text{CVPR'18}}$ [1]	Wide ResNet-38		61.7	63.7
SeeNet $_{\text{NIPS'18}}$ [25]	ResNet-101	✓	63.1	62.8
Zeng *et al* $_{\text{ICCV'19}}$ [61]	DenseNet-169		63.3	64.3
BDSSW $_{\text{ECCV'18}}$ [18]	ResNet-101		63.6	64.5
OAA $_{\text{ICCV'19}}$ [28]	ResNet-101	✓	63.9	65.6
CIAN $_{\text{CVPR'19}}$ [17]	ResNet-101		64.1	64.7
FickleNet $_{\text{CVPR'19}}$ [33]	ResNet-101	✓	64.9	65.3
Ours	ResNet101	✓	**66.1**	**65.9**

In Table 3.3, we show detailed results for each category on the validation set. We compare two groups of results with (bottom) or without (top) applying the CRF [32] refinement to the final segmentation outputs. Compared to the recent FickleNet [33] method that also focuses on improving the initial response map, the proposed algorithm performs favorably for the segmentation task in terms of the mean IoU. We also note that, our results without applying CRF (mIoU as 64.8%) already achieves similar performance compared with the FickleNet (mIoU as 64.9%). In Figure 3.7, we present some examples of the final semantic segmentation results, and show that our results are close to the ground truth segmentation.

3.4.5 Quality of Clustering

Figure 3.9 to Figure 3.11 present exemplar results of the clustering for 3 different classes in the PASCAL VOC 2012 dataset [15] (i.e., class of bird, boat, and sheep). For each class, we show the clustering result of the first round and the third round model, which are the beginning and the final clustering result during the iterative training process.

By observing the visual change between Round-1 and Round-3, we can find the quality of clustering results is enhanced. For example, in Figure 3.9, images of the bird flying in the sky are clustered into different clusters at Round-1, yet such images can be gathered into the same cluster at Round-3. Note that in Figure 3.9 to Figure 3.11, we use yellow boxes to mark the images that have a similar visual style but are clustered into different clusters at Round-1. Whereas images with the coherent visual style are clustered into one cluster at Round-3, which is marked by red boxes.

Results of the final round of clustering present the consistency within a cluster, including size, type, context, and the interaction between objects. The visual change between the first and the last round of clustering demonstrates the effectiveness of our iterative training process, which validates that our learned feature representations are enhanced via the sub-category objective in an unsupervised manner.

3.4.6 Qualitative Comparisons

In this section, we provide additional results for qualitative comparisons, including the visual comparison of initial response prediction and segmentation result. Figure 3.12 presents both initial response maps and segmentation results of the AffinityNet [1] method and ours. We illustrate intermediate results to demonstrate that a better initial seed can benefit the quality of segmentation result, and a number of qualitative examples of our final model are presented in Figure 3.13.

In addition, we also show some failure segmentation cases in Figure 3.8. There are two main issues that would affect the quality of segments: 1) the incompleteness on detailed parts and 2) the ambiguity on object boundaries, which are two challenging problems of the segmentation task. Although there are some failure examples, our

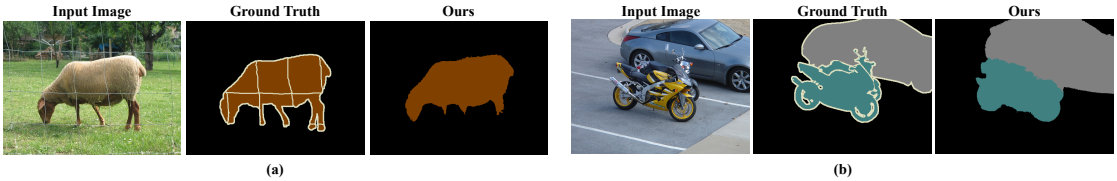

Figure 3.8: Failure semantic segmentation results. (a) Failure cases of the incompleteness on detailed parts. Legs of the animal are missing in the segment. (b) Failure case of the ambiguity on object boundary. There are errors on the boundary region between two objects.

approach can produce high quality semantic segmentation results.

3.5 Summary

In this chapter, we propose a simple yet effective approach to improve the class activation maps by introducing a self-supervised task to discover sub-categories in an unsupervised manner. Without bells and whistles, our approach performs favorably against existing weakly-supervised semantic segmentation methods. Specifically, we develop an iterative learning scheme by running clustering on image features for each parent class and train the classification network on sub-category objectives. Unlike other existing schemes that aggregate multiple response maps, our approach generates better initial predictions without introducing extra complexity or inference time to the model. We conduct extensive experimental analysis to demonstrate the effectiveness of our approach via exploiting the sub-category information. Finally, we show that our algorithm produces better activation maps, thereby improving the final semantic segmentation performance.

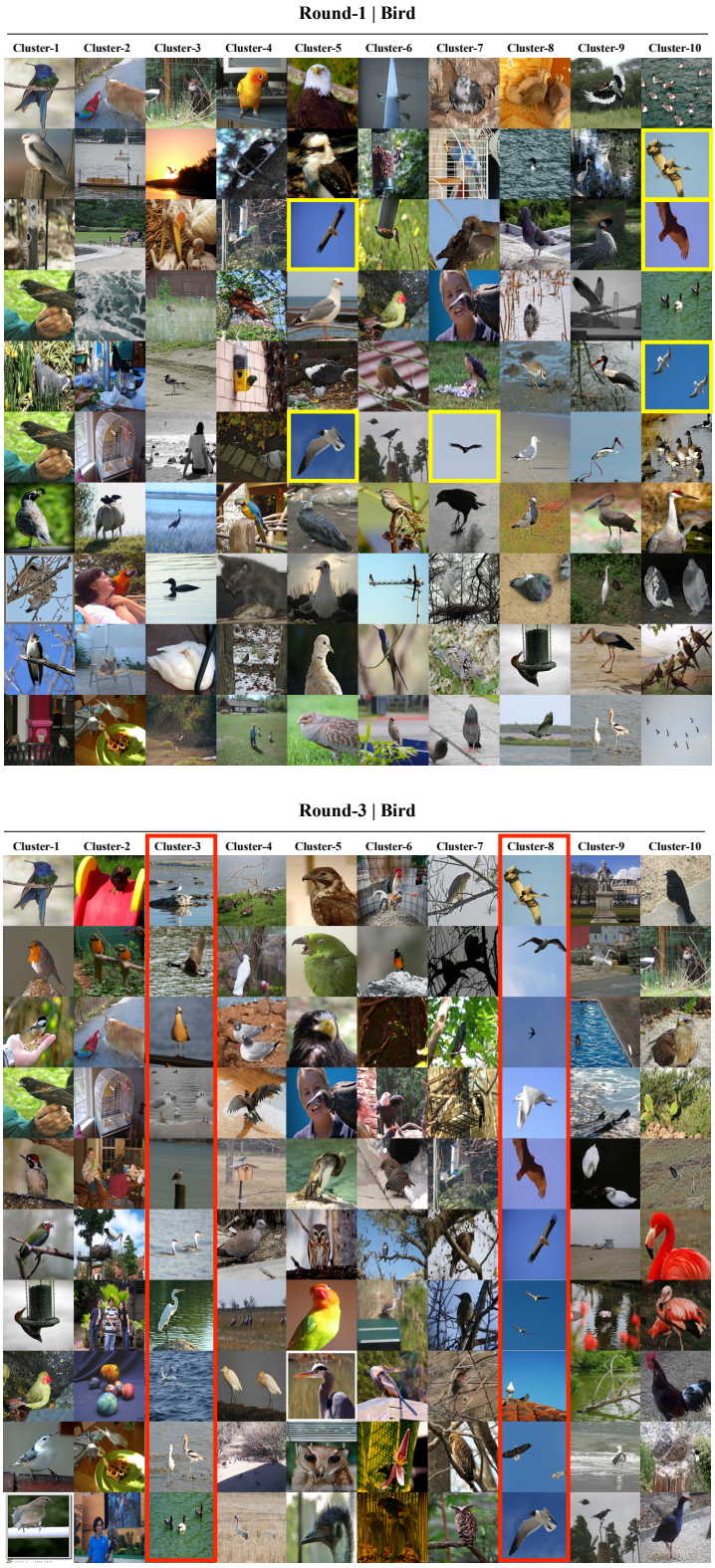

Figure 3.9: Visual comparison of the different rounds of clustering of *bird* class. Red boxes at later round demonstrate sets of image with visual consistency compared to images marked by yellow boxes at the beginning round.

Figure 3.10: Visual comparison of the different rounds of clustering of *boat* class. Images of sailboat are in different clusters at Round-1 while such image can be clustered to one cluster (i.e., Cluster-10) at Round-3.

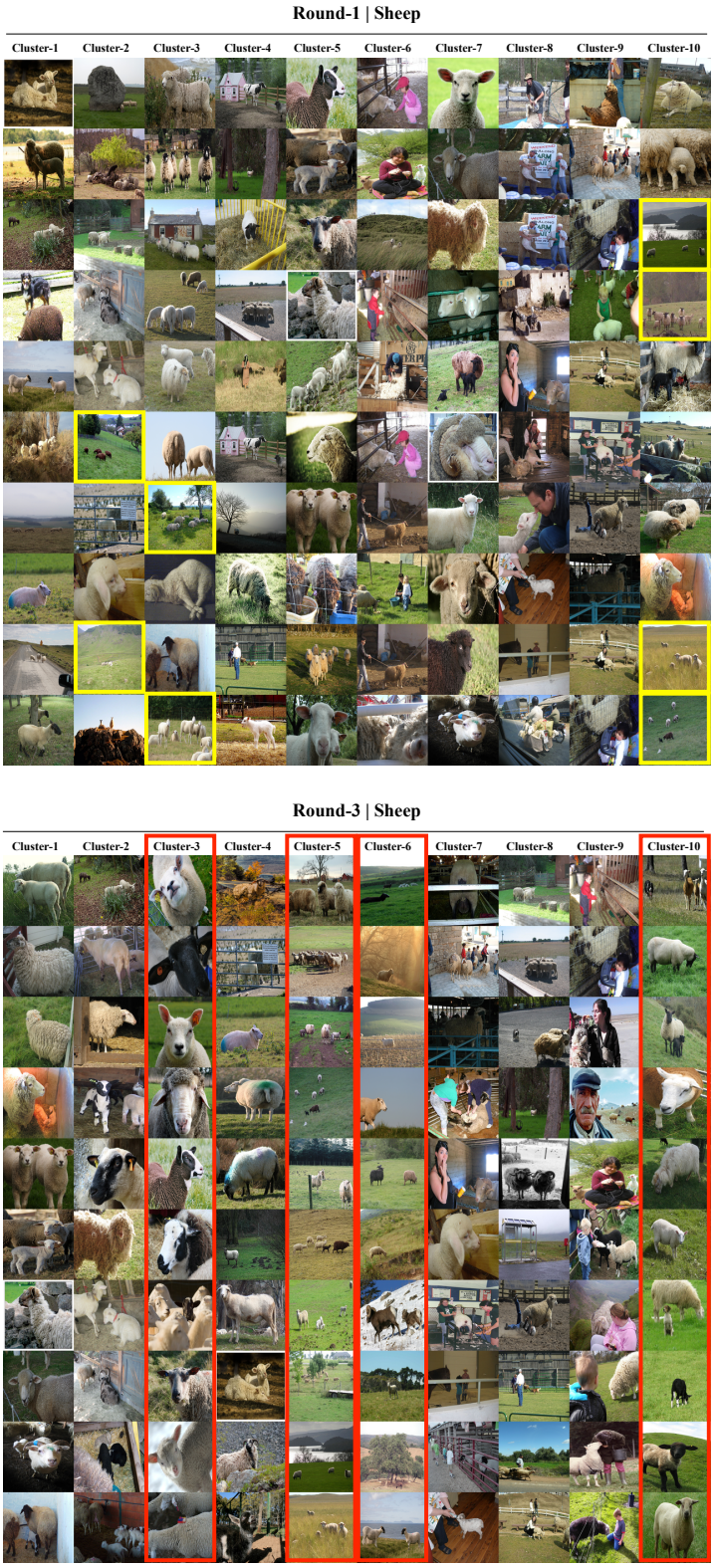

Figure 3.11: Visual comparison of the different rounds of clustering of *sheep* class. Images of sheep in the meadow in distant view are in different clusters at Round-1 while such images can be clustered to one cluster (i.e., Cluster-5) at Round-3.

Figure 3.12: Qualitative comparison of the initial response map and semantic segmentation map. We compare our intermediate and final results with the AffinityNet [1] approach.

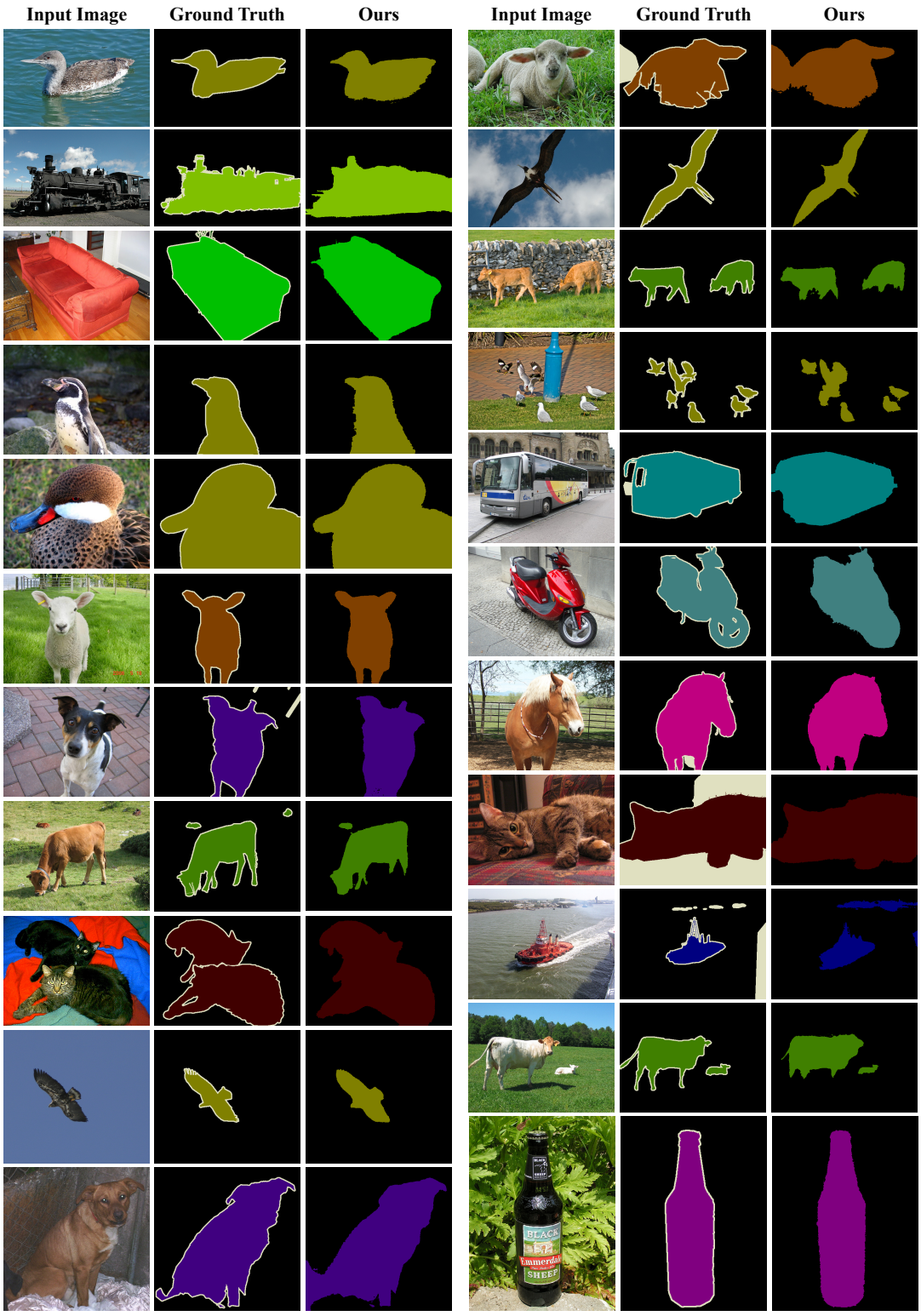

Figure 3.13: Semantic segmentation results on the PASCAL VOC 2012 val images.

Chapter 4

Conclusions and Future Work

In this thesis, we propose two self-regularization methods to improve the localization of object response maps, as an initial step towards weakly-supervised semantic segmentation task using image-level labels. We first present the Mixup-CAM method in Chapter 2, which is an end-to-end trainable network with loss functions designed to systematically control the generation of the response map. Specifically, we adopt the entropy-based loss to regularize the uncertainty, coupled with the mixup data augmentation for producing better response maps on objects. In Chapter 3, we introduce a self-supervised task to discover sub-categories in an unsupervised manner. particularly, we develop an iterative learning scheme by running clustering on image features for each parent class and train the classification network on sub-category objectives. By imposing a more challenging task, the model learns better representations thereby improving the response map. We provide comprehensive analysis of each component in the proposed method and show our approaches produce better activation maps and achieve state-of-the-art performance against existing algorithms. We conclude the thesis by discussing the potential directions for future research.

Large-scale Datasets. To provide more insight and investigate weakly-supervised semantic segmentation, we can first utilize other large-scale datasets such as MS COCO [37] to validate whether our methods could have more advantages. In addition, other scenarios such as outdoor [10] and indoor [66] datasets could be further exploited to observe their different characteristics, and more advanced self-regularization algorithms

can be developed accordingly to deal with various challenging cases.

Other Self-regularization Techniques. Second, we could incorporate other regularization techniques like label smoothing [38], and observe whether it has complementary properties compared to our methods. Moreover, recent advances of contrastive learning [7] also shares a similar concept of learning effective visual representations, and such technique could become meaningful training signals to regularize deep learning models, in which potentially could improve our initial object responses.

Various Training Steps, Annotation Types, and Tasks. Finally, self-regularization methods could be helpful for the other two steps (i.e., response refinement and segmentation model training). In addition, how to integrate these three steps or different kinds of weak supervision signals towards a unified framework is one interesting direction. Beyond weakly-supervised semantic segmentation, improving initial predictions could be helpful for other tasks, such as image retrieval, video analysis, and scene understanding.

With the introduced self-regularization methods in this thesis, we provide a foundation for such future directions towards learning better feature representations, while we target at the weakly-supervised semantic segmentation setting that generates better response maps for object localization as the demonstration task.

Bibliography

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

[2] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei. What's the point: Semantic segmentation with point supervision. In *ECCV*, 2016.

[3] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 2019.

[4] M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

[5] A. Chaudhry, P. K. Dokania, and P. H. Torr. Discovering class-specific pixels for weakly-supervised semantic segmentation. In *BMVC*, 2017.

[6] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *PAMI*, 40, 2017.

[7] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[8] Y.-W. Chen, Y.-H. Tsai, Y.-Y. Lin, and M.-H. Yang. Vostr: Video object segmentation via transferable representations. *IJCV*, 02 2020.

[9] A. Coates and A. Y. Ng. Learning feature representations with k-means. In *Neural networks: Tricks of the trade*, pages 561–580. Springer, 2012.

[10] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[11] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le. Autoaugment: Learning augmentation strategies from data. In *CVPR*, 2019.

[12] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, 2015.

[13] V. R. de Sa. Learning classification with unlabeled data. In *NIPS*, 1994.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[15] M. Everingham, L. J. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[16] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results.

[17] J. Fan, Z. Zhang, and T. Tan. Cian: Cross-image affinity net for weakly supervised semantic segmentation. In *CVPR*, 2019.

[18] R. Fan, Q. Hou, M.-M. Cheng, G. Yu, R. R. Martin, and S.-M. Hu. Associating inter-image salient instances for weakly supervised semantic segmentation. In *ECCV*, 2018.

[19] Y. Grandvalet and Y. Bengio. Semi-supervised learning by entropy minimization. In *NIPS*, 2005.

[20] H. Guo, Y. Mao, and R. Zhang. Mixup as locally linear out-of-manifold regularization. In *AAAI*, 2019.

[21] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In *ICCV*, 2011.

[22] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[23] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[24] J. helhamer, Evan an Long and T. Darrell. Fully convolutional networks for semantic segmentation. *PAMI*, 2016.

[25] Q. Hou, P. Jiang, Y. Wei, and M.-M. Cheng. Self-erasing network for integral object attention. In *NIPS*, 2018.

[26] Z. Huang, X. Wang, J. Wang, W. Liu, and J. Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In *CVPR*, 2018.

[27] W.-C. Hung, V. Jampani, S. Liu, P. Molchanov, M.-H. Yang, and J. Kautz. Scops: Self-supervised co-part segmentation. In *CVPR*, 2019.

[28] P.-T. Jiang, Q. Hou, Y. Cao, M.-M. Cheng, Y. Wei, and H.-K. Xiong. Integral object mining via online attention accumulation. In *CVPR*, 2019.

[29] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *CVPR*, 2017.

[30] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[31] A. Kolesnikov and C. H. Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *ECCV*, 2016.

[32] P. Krahenbuhl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, 2011.

[33] J. Lee, E. Kim, S. Lee, J. Lee, and S. Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *CVPR*, 2019.

[34] K. Li, Z. Wu, K.-C. Peng, J. Ernst, and Y. Fu. Tell me where to look: Guided attention inference network. In *CVPR*, 2018.

[35] S. Lim, I. Kim, T. Kim, C. Kim, and S. Kim. Fast autoaugment. In *NeurIPS*, 2019.

[36] D. Lin, J. Dai, J. Jia, K. He, and J. Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, 2016.

[37] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014.

[38] R. Müller, S. Kornblith, and G. Hinton. When does label smoothing help? In *NIPS*, 2019.

[39] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.

[40] G. Papandreou, L.-C. Chen, K. Murphy, and A. L. Yuille. Weakly-and semi-supervised learning of a dcnn for semantic image segmentation. In *ICCV*, 2015.

[41] D. Pathak, P. Krahenbuhl, and T. Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *ICCV*, 2015.

[42] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[43] M. Paulin, M. Douze, Z. Harchaoui, J. Mairal, F. Perronin, and C. Schmid. Local convolutional features with unsupervised training for image retrieval. In *CVPR*, 2015.

[44] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton. Regularizing neural networks by penalizing confident output distributions. In *ICLR Workshop*, 2017.

[45] P. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015.

[46] S. Sarfraz, V. Sharma, and R. Stiefelhagen. Efficient parameter-free clustering using first neighbor relations. In *ICCV*, 2019.

[47] W. Shimoda and K. Yanai. Self-supervised difference detection for weakly-supervised semantic segmentation. In *ICCV*, 2019.

[48] K. K. Singh and Y. J. Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*, 2017.

[49] C. Summers and M. J. Dinneen. Improved mixed-example data augmentation. In *WACV*, 2019.

[50] S. Thulasidasan, G. Chennupati, J. Bilmes, T. Bhattacharya, and S. Michalak. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. In *NeurIPS*, 2019.

[51] Y.-H. Tsai, G. Zhong, and M.-H. Yang. Semantic co-segmentation in videos. In *ECCV*, 2016.

[52] L. J. P. van der Maaten and G. E. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[53] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, 2019.

[54] P. Vernaza and M. Chandraker. Learning random-walk label propagation for weakly-supervised semantic segmentation. In *CVPR*, 2017.

[55] X. Wang, S. You, X. Li, and H. Ma. Weakly-supervised semantic segmentation by iteratively mining common object features. In *CVPR*, 2018.

[56] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *CVPR*, 2017.

[57] Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, and T. S. Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *CVPR*, 2018.

[58] Z. Wu, C. Shen, and A. Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 90:119–133, 2019.

[59] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016.

[60] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.

[61] Y. Zeng, Y. Zhuge, H. Lu, and L. Zhang. Joint learning of saliency detection and weakly supervised semantic segmentation. In *CVPR*, 2019.

[62] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018.

[63] X. Zhang, Y. Wei, J. Feng, Y. Yang, and T. S. Huang. Adversarial complementary learning for weakly supervised object localization. In *CVPR*, 2018.

[64] G. Zhong, Y.-H. Tsai, and M.-H. Yang. Weakly-supervised video scene co-parsing. In *ACCV*, 2016.

[65] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016.

[66] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ADE20K dataset. *CoRR*, abs/1608.05442, 2016.