BLPSeg: Balance the Label Preference in Scribble-Supervised Semantic Segmentation

Yude Wang, *Student Member, IEEE*, Jie Zhang, *Member, IEEE*, Meina Kan, *Member, IEEE*, Shiguang Shan, *Fellow, IEEE*, and Xilin Chen, *Fellow, IEEE*

***Abstract*— Scribble-supervised semantic segmentation is an appealing weakly supervised technique with low labeling cost. Existing approaches mainly consider diffusing the labeled region of scribble by low-level feature similarity to narrow the supervision gap between scribble labels and mask labels. In this study, we observe an annotation bias between scribble and object mask, *i.e.,* label workers tend to scribble on the spacious region instead of corners. This label preference makes the model learn well on those frequently labeled regions but poor on rarely labeled pixels. Therefore, we propose BLPSeg to balance the label preference for complete segmentation. Specifically, the BLPSeg first predicts an annotation probability map to evaluate the rarity of labels on each image, then utilizes a novel BLP loss to balance the model training by up-weighting those rare annotations. Additionally, to further alleviate the impact of label preference, we design a local aggregation module (LAM) to propagate supervision from labeled to unlabeled regions in gradient backpropagation. We conduct extensive experiments to illustrate the effectiveness of our BLPSeg. Our single-stage method even outperforms other advanced multi-stage methods and achieves state-of-the-art performance.**

***Index Terms*— Scribble-supervised, weakly supervised, semantic segmentation.**

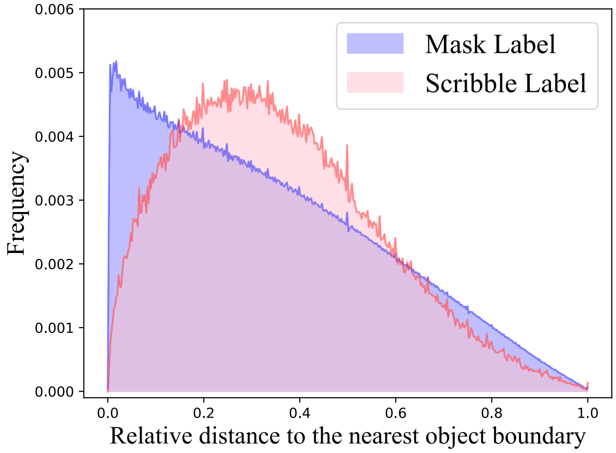


Fig. 1. The distribution of relative distances from the labeled pixels to their nearest object boundary. The relative distance of each labeled pixel is normalized by the maximum distance of the corresponding category in the image, which removes the interference of object size.

I. Introduction

SEMANTIC segmentation is a fundamental computer vision task that assigns pixel-level class labels for each image. Since the deep learning era, it has achieved many promising breakthroughs [1], [2], [3], [4], [5], [6], [7]. However, the fully supervised semantic segmentation methods heavily rely on a large amount of pixel-level annotations, which significantly raises the cost in practical applications. Therefore, recent works turn to tackle training semantic segmentation models by weak labels, *e.g.,* bounding boxes [8], [9], [10], [11], [12], scribbles [13], [14], [15], [16], [17], points [18], [19], and image-level class labels [20], [21], [22], [23], [24], which significantly reduce the annotation cost. In this study, we mainly focus on scribble-supervised semantic segmentation.

Although scribble is one of the most popular annotation formats for semantic segmentation, a supervision gap exists between scribble and the ideal segmentation mask. Firstly, scribble only labels a small part of pixels and remains massive pixels unlabeled. Most existing methods [15], [16], [25] narrow the supervision gap from this perspective, utilizing local similarity to diffuse the labeled region of scribble labels.

However, we find there is another kind of supervision gap. In Fig. 1, the statistic result shows that the position distributions of scribble and mask labels are pretty different. The reason is that annotation workers prefer to draw a scribble in the central area of the object, leading to an annotation bias. Fig. 2 explicitly illustrates label preference between scribble and mask. This label preference makes the model learn well in those frequently labeled regions but poor in other rarely labeled ones, impairing the training of the model to approach the performance achieved in a fully supervised manner.

Manuscript received 4 November 2022; revised 13 June 2023; accepted 19 July 2023. Date of publication 21 August 2023; date of current version 6 September 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2021YFC3310100; and in part by the National Natural Science Foundation of China under Grant 62176251, Grant 62122074, and Grant 61976219. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ming-Ming Cheng. *(Corresponding author: Jie Zhang.)*

Yude Wang, Jie Zhang, Meina Kan, and Xilin Chen are with the Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: yude.wang@vipl.ict.ac.cn; zhangjie@ict.ac.cn; kanmeina@ict.ac.cn; xlchen@ict.ac.cn).

Shiguang Shan is with the Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China, also with the University of Chinese Academy of Sciences, Beijing 100049, China, and also with the Peng Cheng National Laboratory, Shenzhen 518055, China (e-mail: sgshan@ict.ac.cn).

Digital Object Identifier 10.1109/TIP.2023.3301342

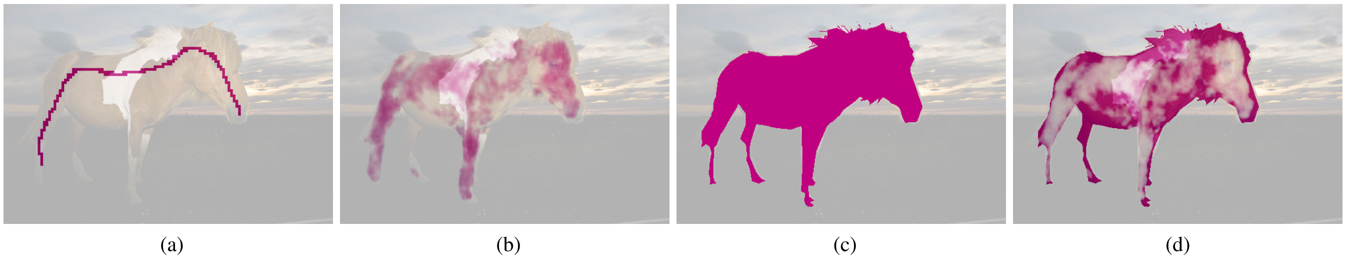


Fig. 2. Illustration of annotation bias. (a) Scribble label. (b) Scribble annotation probability map, which highlights the regions easily being labeled by scribble. (c) Mask label. (d) The difference map between (c) and (b), which highlights the region rarely labeled by scribbles.

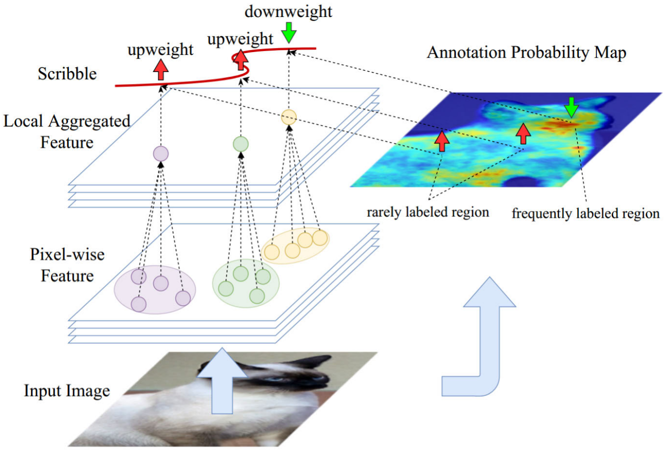


Fig. 3. The overview of our proposed BLPSeg. The BLPSeg is designed to balance label preference in two views: (1) BLPSeg predicts an annotation probability map to reveal label preference directly and dynamically adjusts the pixel-wise training weights. (2) BLPSeg aggregates feature in the local range to expand supervision from labeled to unlabeled pixels, which indirectly alleviates the annotation bias.

In this study, we propose a BLPSeg method to balance the label preference for more accurate segmentation with scribble labels. As shown in Fig. 3, the BLPSeg first estimates the annotation probability of each pixel being labeled by scribbles, revealing the spatial label preference as given in Fig. 2b. Then we introduce a novel BLP loss to highlight those rare annotations, which lie in the low activation region of the annotation probability map. Specifically, BLP loss assigns a relative down-weighted and up-weighted loss to the frequently and rarely labeled pixels. Thus it alleviates the negative impact caused by annotation bias and balances the model training on the whole image.

Besides, we embed a simple but effective local aggregation module (LAM) in BLPSeg to further improve the segmentation prediction. The LAM follows previous methods that diffuse the supervision from labeled pixels. Concretely, as shown in Fig. 3, the LAM aggregates the high-level features of pixels by their low-level similarity in the local range. Thus the supervision can be propagated from labeled pixels to more unlabeled pixels correspondingly in the gradient backpropagation.

Extensive experiments are conducted on PASCAL VOC 2012 and PASCAL-Context datasets to demonstrate the effectiveness of our BLPSeg. Our single-stage method even outperforms many multi-stage approaches, achieving a new state-of-the-art result for scribble-supervised semantic segmentation.

Overall, our contributions are summarized as follows:

1) We propose a BLPSeg method to balance the label preference in scribble-supervised semantic segmentation.
2) The BLPSeg introduces a novel BLP loss to balance the learning weights of pixels with different annotation probabilities, and embeds another local aggregation module (LAM) to propagate supervision for better training.
3) Extensive experiments illustrate that our single-stage method achieves a new state-of-the-art performance on both PASCAL VOC 2012 and PASCAL-Context datasets.

II. Related Works

Semantic Segmentation is a fundamental computer vision task achieving remarkable progress in the last decade [1], [2], [3]. This section briefly revisits fully supervised, weakly supervised semantic segmentation, and unsupervised image segmentation methods.

A. Fully Supervised Semantic Segmentation

The deep neural network has become the mainstream method for computer vision tasks since AlexNet [26] achieves significant breakthroughs in image classification. Compared to image classification making a global prediction, semantic segmentation is a more challenging task that makes a pixel-level prediction on images.

The development of semantic segmentation can be divided into three stages since the deep learning era. In the first stage, there are two basic strategies to improve model by enlarging the receptive field [2] of neural network and leveraging multi-layer information [3], [27]. Secondly, researchers focus on exploring context information for the discrimination of each pixel. The self-attention module [28] and its various low-rank form [4], [29] dominated the period. In the current third stage, many segmentation works [30], [31] are proposed based on vision transformer [32] and large-scale unsupervised pretraining [33], [34], which also contribute to a series of milestone achievements in many computer vision tasks. Moreover, there is a tendency to merge semantic segmentation, instance segmentation, and panoptic segmentation by a unified model [35] thanks to the bipartite matching proposed by DETR [36]. In this study, we will design our scribble-supervised segmentation model based on these development experiences of fully supervised semantic segmentation.

B. Weakly Supervised Semantic Segmentation

Compared to fully supervised semantic segmentation, weakly supervised semantic segmentation has attracted many research interests attributed to its lower annotation costs. Recent works significantly narrow the performance gap to the fully supervised methods based on various weak supervision, *e.g.,* image-level classification labels [20], [21], [22], [23], [37], [38], bounding boxes [8], [9], [10], [11], points [18], [39], and scribbles [13], [14], [15], [16], [17], [25], [40]. Based on these weakly supervised learning methods, integrating noise label learning [41] can further improve segmentation prediction. This study mainly discusses the semantic segmentation methods supervised by scribble annotation.

The approaches for scribble-supervised semantic segmentation can be divided into two groups. The first method aims to expand scribbles and improve the quality of pixel-level pseudo labels, also named the label diffusion strategy. ScribbleSup [13] leverages super-pixels to expand initial scribble labels coverage. RAWKS [14] expands sparse annotation via random-walk hitting probability. Besides, some works refine feature maps to serve as self-training pseudo labels. DenseCRF [2] is a traditional approach to align the counter of the predicted feature map to the image boundary, and it is widely used as a post-processing step in semantic segmentation. BPG [42] iteratively refines the feature map with the help of an additional boundary detector.

The second kind of scribble-supervised semantic segmentation method mainly focuses on introducing additional similarity regularization loss for network training. Deriving from the traditional DenseCRF [2], various graphical regularization losses [15], [16], [43] are proposed to cluster pixel features according to their low-level similarity, *i.e.,* color, position, and texture. Besides, SPML [40] presents four types of contrastive relation to cluster similar pixel features to learn distinct representation. Moreover, Pan et al. [44] regularize network training implicitly by proposing the eigenspace consistency of the feature map inner product matrix from different views. Although additional similarity regularization narrows the supervision gap between scribble and mask labels, the model training is sensitive to the weight of additional loss items and easily falls into a trivial solution.

C. Unsupervised Image Segmentation

Unsupervised Image segmentation is a fundamental preprocessing step for many vision algorithms, especially for the limited annotation regime. It first reduces subsequent algorithms' complexity by clustering pixels into several segments. Traditional image segmentation approaches [45], [46], [47], [48] usually leverage clustering algorithm [49], [50], [51] iteratively update the segment clusters according to the pixels' coordinates and manually designed visual feature. Spectral graph theory [52] transfers the individual representation of each pixel into a densely connected graph so that the segmentation results consider the feature distribution of all pixels. Based on the graph, a series of graph cut works [53], [54] build additional nodes and apply the minimum cut algorithm on the graph to find the optimal segment boundary, which also can be designed as interactive methods to segment images according to the clicks of users. Although these traditional segmentation approaches may not capture semantic information, they work well to get several segments with low intra-variance.

Unsupervised image segmentation also has a breakthrough due to the development of self-supervised image representation learning. Most works [55], [56], [57], [58] focus on training the network by a pretext task to get a discriminative representation. IIC [55] augments the image into multi-views and enforces their representation to be the same. The feature representation achieved by IIC is highly aligned with semantic classes and performs promising segments after clustering. DINO [56] adopts a similar learning objective with IIC. However, the segmentation step has already been embedded in the vision transformer network [32] where the attention map of the class token reveals the corresponding segmentation region. InfoSeg [59] dynamically compresses pixel-level features into limited prototypes in the process of feature map reconstruction, and thus segments the pixels into multiple regions according to their prototypes. Compared to the traditional image segmentation approaches, deep learning based unsupervised image segmentation methods capture high-level information from large-scale datasets. Therefore, they achieve more promising segmentation results than traditional ones but still need to improve in some complex scenes.

III. Approach

In this section, we introduce our proposed BLPSeg in detail. The BLPSeg is designed to tackle the challenge of annotation bias shown in Fig. 1. The architecture of BLPSeg is given in Fig. 4. The main stem of BLPSeg is attached by two additional heads: scribble head and segmentation head. The scribble head is utilized to predict an annotation probability map to reveal the label preference of each image (Sec. III-A). The segmentation head is used for segmentation prediction, followed by a novel local aggregation module (LAM). The LAM is designed to aggregate pixel-wise features by local similarity and propagate supervision from labeled to larger unlabeled regions in gradient backpropagation (Sec. III-B). Finally, we introduce a novel BLP loss for segmentation head training, which uses the annotation probability map predicted by scribble head to reveal the rarity of scribble labels and balance label preference in the training process (Sec. III-C).

A. Annotation Probability Map

It is crucial to estimate the label preference on each image first to alleviate the negative impact caused by annotation bias. Therefore, we design a scribble head in Fig. 4 to estimate the annotation probability map of the scribble label.

Suppose each image has N pixels, and there are K classes annotated by scribble label $y \in \{0, 1\}^{K \times N}$. The scribble label y satisfying $\sum_{c=1}^{K} y_{c,n} = 1$ for each labeled pixel, and $\sum_{c=1}^{K} y_{c,n} = 0$ for each unlabeled pixel. Then the scribble head introduces an additional $\varnothing$ class to the whole unlabeled pixels, and applies a weighted cross entropy loss for

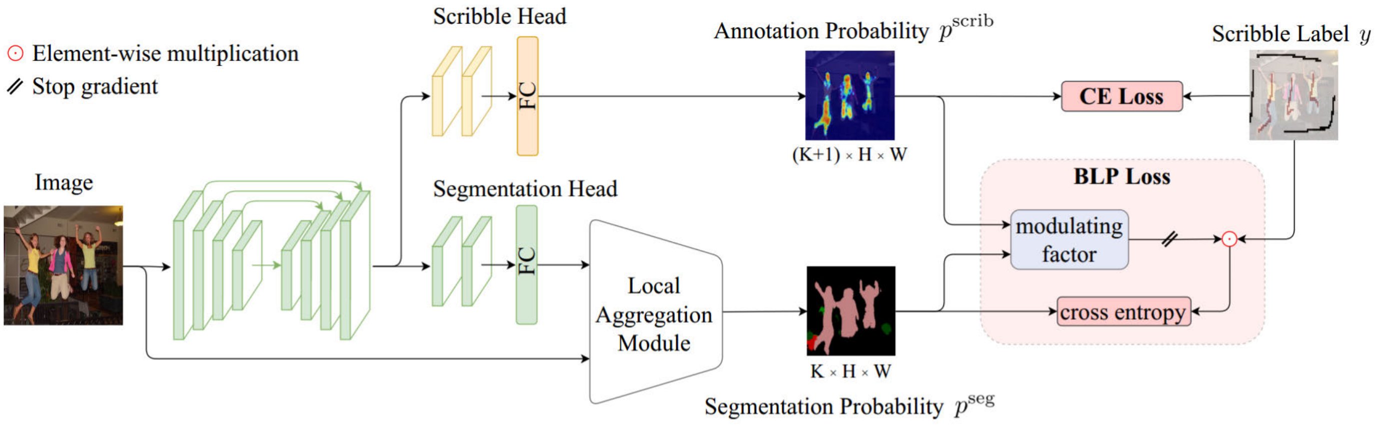
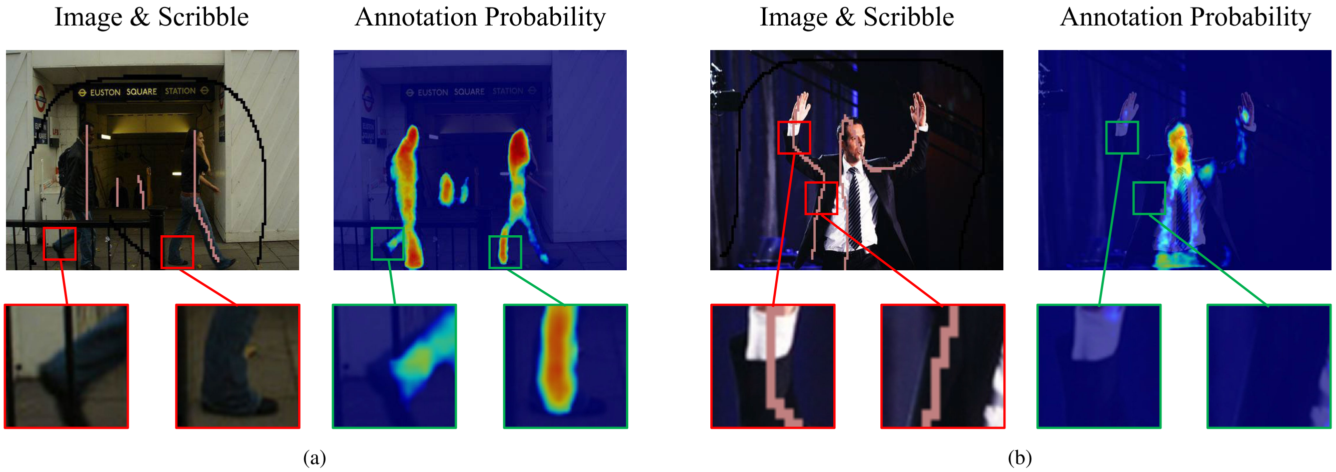


Fig. 4. The architecture of BLPSeg. (i) Scribble head predicts annotation probability p^{scrib} to reveal the label preference. (ii) The output of local aggregation module (LAM) p^{seg} serves as final prediction. (iii) Modulating factor in BLP loss up-weights rare labels to alleviate annotation bias during model training.



(a)

(b)

Fig. 5. The comparison of annotation probability map and scribble labels. The annotation probability map highlights the pixels most likely to be labeled by scribble, although sometimes it is not labeled actually, such as the bounding box in (a). On the contrary, when the scribbles label on the pixels having low annotation probability, it means this scribble is a rare annotation and should be focused on during training, *e.g.,* the cases given in (b).

pixel-level classification:

$$WCE(p^{\text{scrib}}, y') = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{K+1} w_c y'_{c,n}\log(p^{\text{scrib}}_{c,n}), \qquad (1)$$

where $p^{\text{scrib}} \in \mathbb{R}^{(K+1)\times N}$ is the estimated annotation probability of scribble. $y' \in \{0,1\}^{(K+1)\times N}$ is the pixel-level classification label with $\varnothing$ class, which sastifying:

$$y'_{c,n} = \begin{cases} y_{c,n}, & \text{if} \quad c \le K \\ 1, & \text{if} \quad c = K+1 \quad \text{and} \quad \sum_{c=1}^{K} y_{c,n} = 0 \\ 0, & \text{if} \quad c = K+1 \quad \text{and} \quad \sum_{c=1}^{K} y_{c,n} = 1 \end{cases} \qquad (2)$$

And w is the class weight to balance the training of each class. All the weights set as 1 except w_{K+1} for $\varnothing$ class, which set smaller to balance the vast number of unlabeled pixels. With the additional $\varnothing$ class, the "unlabeled" cases are considered into classification and equal to other specific categories. And the prediction can be described as the $K+1$ classification probability, in other words, the probability of pixels labeled by scribbles or not.

The annotation probability map can be regarded as the annotation position prior for each image. As shown in Fig. 5, the annotation probability map highlights the region most likely to be labeled, although sometimes it is not labeled by a single scribble, *e.g.,* the foot in Fig. 5a. Besides, the low annotation probability regions sometimes also are labeled, *e.g.,* the hand region in Fig. 5b, which is rare and should be focused on during training to balance label preference.

B. Local Aggregation Module

Label diffusion is a classical strategy to expand labeled regions in scribble-supervised semantic segmentation. It is also an indirect method to balance label preference to some extent. However, label diffusion generally adopts super-pixels or other manually designed multi-stage rules to expand scribble, which inevitably introduces some noise and may hurt the network training. In our BLPSeg, we propose a simple but effective local aggregation module (LAM). The LAM leverages a gradient diffusion strategy to expand supervision and is jointly optimized with the segmentation network, which differs from label diffusion. The LAM is formulated as follows:

$$\mathbf{X}^{\text{out}} = \mathbf{X}^{\text{in}}\mathbf{A}. \qquad (3)$$

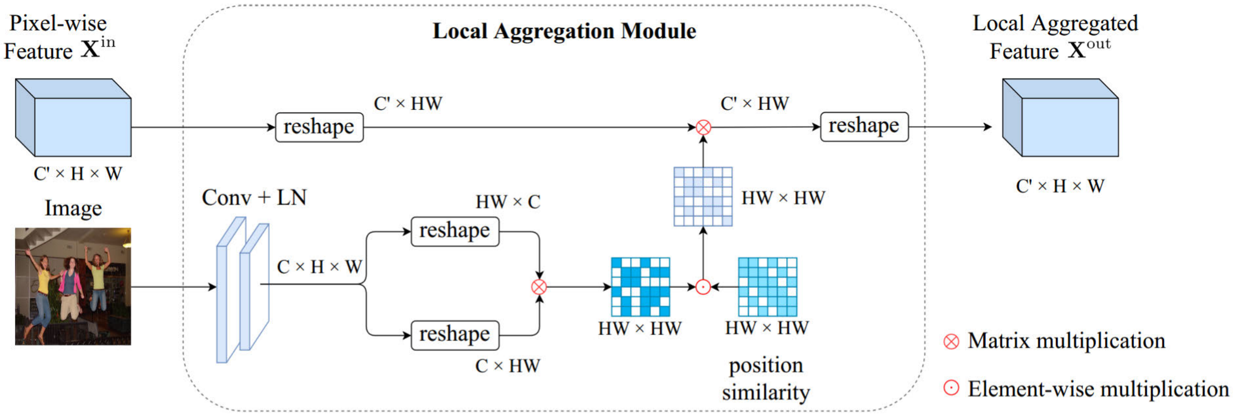


Fig. 6. The overview of local aggregation module (LAM). In the forward pass, the LAM aggregates the pixel-wise features by their low-level similarity which is learned adaptively. Therefore, the LAM can diffuse supervision from labeled to unlabeled pixels in backward propagation.

Here $\mathbf{X}^{\text{in}} \in \mathbb{R}^{D\times N}$ is the D dimensional input feature of LAM for N pixels. $\mathbf{X}^{\text{out}} \in \mathbb{R}^{D\times N}$ is the output feature of LAM, which is locally aggregated from $\mathbf{X}^{\text{in}}$. $\mathbf{A} \in \mathbb{R}^{N\times N}$ is a similarity matrix for feature aggregation and defined as:

$$\mathbf{A}_{i,j} = \frac{P_{\text{low}}(i,j)\cdot P_{\text{dis}}(i,j)}{\sum_k P_{\text{low}}(k,j)\cdot P_{\text{dis}}(k,j)}, \tag{4}$$

$$P_{\text{low}}(i,j) = \frac{e^{\mathbf{f}_i^\top \mathbf{f}_j}}{\sum_k e^{\mathbf{f}_k^\top \mathbf{f}_j}}, \quad P_{\text{dis}}(i,j) = \exp(-\frac{\parallel \mathbf{s}_i - \mathbf{s}_j \parallel^2}{2\sigma^2}), \tag{5}$$

where $P_{\text{low}}(i,j)$ is the low-level feature similarity between pixel i and j, $P_{\text{dis}}(i,j)$ is a distance kernel to regularize feature aggregation in the local range. $\mathbf{f}$ indicates the low-level feature and $\mathbf{s}$ is the spatial position of pixel. Hyperparameter σ controls the distance regularization range.

We formulate the gradient of LAM to introduce its gradient diffusion process:

$$\frac{\partial\mathcal{L}}{\partial\mathbf{X}^{\text{in}}} = \frac{\partial\mathcal{L}}{\partial\mathbf{X}^{\text{out}}}\mathbf{A}^\top. \tag{6}$$

Here $\mathcal{L}$ is the training loss of the network. More specifically, the features of LAM can be written as the feature concatenation from N_l labeled pixels and N_u unlabeled pixels as $\mathbf{X}^{\text{in}} = (\mathbf{X}_l^{\text{in}}, \mathbf{X}_u^{\text{in}})$ and $\mathbf{X}^{\text{out}} = (\mathbf{X}_l^{\text{out}}, \mathbf{X}_u^{\text{out}})$. Because $\mathcal{L}$ only computes on labeled pixels and $\frac{\partial\mathcal{L}}{\partial\mathbf{X}_u^{\text{out}}} = 0$, the Eq.(6) can be rewritten as:

$$\left(\frac{\partial\mathcal{L}}{\partial\mathbf{X}_l^{\text{in}}}\ \frac{\partial\mathcal{L}}{\partial\mathbf{X}_u^{\text{in}}}\right) = \left(\frac{\partial\mathcal{L}}{\partial\mathbf{X}_l^{\text{out}}}\ 0\right)\mathbf{A}^\top. \tag{7}$$

Therefore, the supervision from N_l labeled pixels is propagated to both labeled and unlabeled pixels by LAM during network gradient backpropagation, contributing to the segmentation model learning with scribble labels.

The gradient diffusion of our LAM is a learnable process compared to the classical label diffusion strategy which relies on manually designed diffusion rules, *e.g.,* super-pixels. The similarity $\mathbf{A}$ in LAM dynamically summarizes the low-level representation between various classes, making the diffusion process more adaptively.

According to Eq. (3), the LAM should be attached to the network feature before the last fully connected layer. But it is equivalent to place LAM before or after fully connected layer:

$$\mathbf{Z} = \mathbf{W}_{\text{FC}}{}^\top\left(\mathbf{X}^{\text{in}}\mathbf{A}\right) = \left(\mathbf{W}_{\text{FC}}{}^\top\mathbf{X}^{\text{in}}\right)\mathbf{A}, \tag{8}$$

where $\mathbf{W}_{\text{FC}} \in \mathbb{R}^{D\times K}$ is the parameter of the fully connected layer. Generally, the feature dimension D is much larger than the class number K. Therefore, we place LAM after the fully connected layer in practice for less computation cost. We will introduce more implementation details in Sec. IV-B.

C. BLP Loss

There are two strategies to change the distribution learned by the model: resampling the input sample and reweighting the training loss of each sample. In our BLPSeg, we follow the second strategy and introduce a BLP loss to tackle the challenge of annotation bias. The BLP loss is formulated as follows:

$$BL(p^{\text{seg}}, p^{\text{scrib}}, y) = -\frac{1}{\mathcal{N}_{\text{seg}}}\sum_{n=1}^{N}\sum_{c=1}^{K} y_{c,n}\delta(p_{c,n}^{\text{scrib}}, p_{c,n}^{\text{seg}}) \times \log(p_{c,n}^{\text{seg}}), \tag{9}$$

$$\mathcal{N}_{\text{seg}} = \sum_{n=1}^{N}\sum_{c=1}^{K} y_{c,n}, \quad p_{c,n}^{\text{seg}} = \text{softmax}(\mathbf{Z}_{c,n}), \tag{10}$$

$$\delta(p_{c,n}^{\text{scrib}}, p_{c,n}^{\text{seg}}) = (1 - p_{c,n}^{\text{scrib}} p_{c,n}^{\text{seg}})^\gamma. \tag{11}$$

Here $p_{c,n}^{\text{seg}} \in \mathbb{R}^{K\times N}$ is the predicted segmentation probability of class c on position n by segmentation head in Fig. 4, which is normalized over all classes by softmax function from the LAM output $\mathbf{Z} \in \mathbb{R}^{K\times N}$(Eq. 8). $p_{c,n}^{\text{scrib}} \in \mathbb{R}^{(K+1)\times N}$ is the scribble annotation probability predicted by scribble head, and $y \in \{0,1\}^{K\times N}$ is the scribble label. Noting that $\varnothing$ class probability in p^{scrib} is not leveraged in BLP loss. $\mathcal{N}_{\text{seg}}$ denotes the number of pixels labeled by scribble in the image. Additionally, a hyperparameter γ is introduced to smoothly adjusts the modulating factor δ for each labeled pixel in training.

The design of BLP loss derives from the *Focal Loss* [60], which aims to tackle class imbalance in object detection. But BLP loss is more adaptive to our specific problem. In BLP loss, when a pixel is labeled by scribble that $y_{c,n} = 1$ with a low annotation probability $p^{\text{scrib}}_{c,n} \to 0$, the modulating factor $\delta \to 1$ preserves a relatively large weight for these labeled pixels, which is rarely labeled and should be focused on.

Besides, the modulating factor is designed as $(1 - p^{\text{scrib}}_{c,n} p^{\text{seg}}_{c,n})^{\gamma}$ instead of $(1 - p^{\text{scrib}}_{c,n})^{\gamma}$ to avoid all the pixels with high annotation probability $p^{\text{scrib}}_{c,n} \to 1$ being assigned a relatively small loss without distinction. And the loss degenerates to the form of focal loss when $p^{\text{scrib}}_{c,n} \to 1$.

Finally, we summarize the overall loss for BLPSeg training as follows:

$$\mathcal{L} = WCE(p^{\text{scrib}}, y') + BL(p^{\text{seg}}, p^{\text{scrib}}, y), \tag{12}$$

where the first item is used for scribble head training to predict annotation probability map p^{scrib}, and the second item provides supervision for segmentation head learning. In the following section, we perform a series of experiments to assess the effectiveness of each component of BLPSeg.

IV. Experimental Settings

A. Datasets

Following many previous scribble-supervised semantic segmentation works [13], [14], [15], [16], in this paper, we adopt two protocols to evaluate our method, which is introduced by PASCAL Scribble [13]. The first protocol is based on the augmented PASCAL VOC 2012 dataset [61]. There are 10582 images with scribble labels for training and 1449 images with mask labels for validation. The dataset contains 20 foreground categories with another background category. The second protocol is based on PASCAL-Context dataset. There are 4998 training images with scribble labels and 5105 validation images with mask labels in PASCAL-Context. The dataset contains 59 classes including both object and stuff, with another background class. The scribble annotations derive from PASCAL-Scribble [13], and the official PASCAL VOC dataset provides a validation set with pixel-level ground truth. Mean intersection over union (mIoU) is utilized as a metric to evaluate segmentation results on both datasets, which is the same as the fully supervised semantic segmentation setting.

B. Implementation Details

In the training period, image samples are randomly rescaled by [0.75, 1.25] and randomly cropped by 384×384. We utilize the classical encoder-decoder structure as our basic framework and adopt ResNet101 [62] as the encoder. The ResNet101 is pretrained on ImageNet [63] for weight initialization. The output feature of the encoder is fed into the PPM [3] module and then aggregated with the 4 layers of the encoder feature to serve as the input of the UperNet [64] decoder. The encoder-decoder is followed by the scribble head and segmentation head, both implemented by two layers "1 × 1 Conv + BN + ReLU" structure without weight-sharing. And the final classification layers are fully connected layers with $K + 1$ and K dimensional output, respectively. Besides, The w_{K+1} for $\varnothing$ class in scribble head is set as 0.02 in all experiments to balance the vast number of unlabeled pixels.

TABLE I

Ablation Studies for Each Part of Our Method. All the Models Are Evaluated on PASCAL VOC 2012 *val* Set Without CRF

LAM	BLP Loss	mIoU(%)				
		1	2	3	mean	std
-	-	72.839	72.556	72.528	72.641	0.172
✓	-	73.798	74.004	74.242	74.015	0.222
✓	✓	75.604	75.663	75.659	**75.642**	0.033

As for LAM, we leverage a 4 × 4 convolution layer followed by layer normalization [65] to extract low-level features from input images. The convolution layer performs down-sampling directly by a stride of 4 to align with the resolution of segmentation head predictions for the feature aggregation.

We adopt AdamW optimizer for our model training, with an initial learning rate of $lr_{init} = 0.0003$, weight decay of 0.01, and employ poly policy $lr_{itr} = lr_{init}(1 - \frac{itr}{itr_{\max}})^{\gamma}$ with $\gamma = 0.9$ for learning rate decay. The self-training strategy is introduced at 10k step, replacing the supervision of segmentation head by its prediction p^{seg} which is sharpened into one-hot form. In our experiments, we leverage the automatic mixed precision package provided by PyTorch [66] to reduce GPU memory usage and boost training speed. The network is trained on two NVIDIA V100 GPUs for 40 epochs with a batch size of 10. During inference, the scribble head is abandoned. The output of LAM is regarded as the model's final prediction. All the models are tested by single-scale except the results in Tab. VII and Tab. VIII, which follows the most recent works [25], [67] and adopted multi-scale and horizontal flip augmentation strategy for fair comparison during model testing.

V. Experimental Results

A. Ablation Study

We first conduct ablation studies on PASCAL VOC 2012 validation set, where only scribble labels are available following [13], to reveal the effectiveness of each component in our proposed method. We run each setting three times and report the mean result. As shown in Tab. I, the local aggregation module brings a 1.374% ↑ mIoU performance gain compared to the baseline firstly, and training with BLP loss further increases the performance by 1.627% ↑ mIoU. It illustrates that every component plays an essential role in our BLPSeg.

B. Label Preference Analysis

In our approach, the prediction of the annotation probability map plays a crucial role in BLP loss. The annotation probability map is expected to reveal label preference for each image in BLP loss. However, each image sample only has scribble labels instead of a label preference ground truth, making it hard to evaluate the quality of the predicted annotation probability map. We find a way to qualitatively compare the predicted annotation probability map and real label preference.

We draw the distribution of relative distance from each pixel to their nearest object boundary, as the same as Fig. 1.

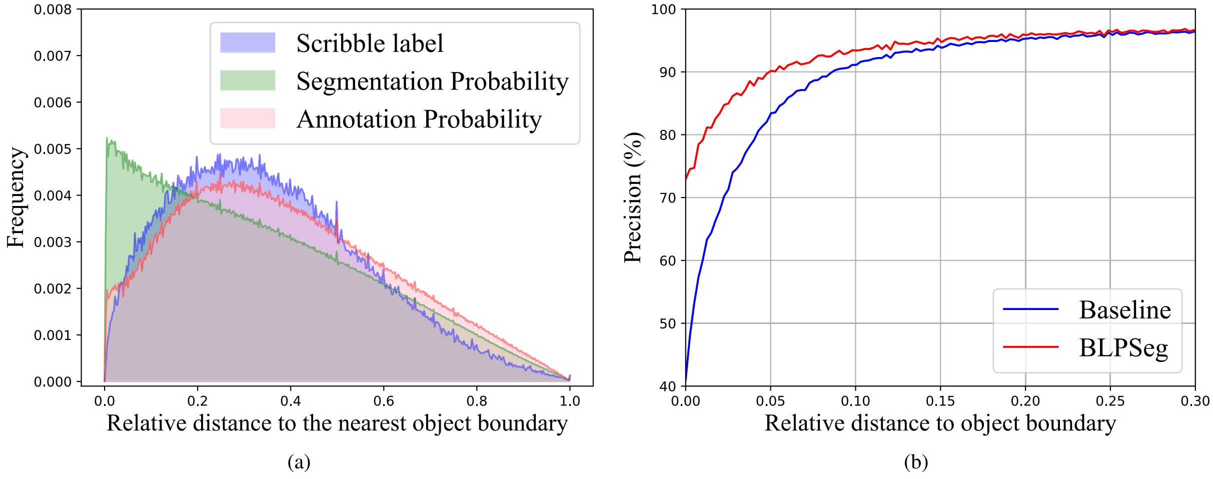


Fig. 7. (a) The distribution of relative distances from high confident pixels to their nearest object boundary. The distribution of scribble label is consistent with Fig. 1. (b) Precision for pixels with different relative distances to the nearest object boundary.

We formulate the probability distribution as follows:

$$P(X) = \frac{\sum_n \lambda_n \cdot \mathbb{1}(X = d_n)}{\sum_n \lambda_n}. \tag{13}$$

$$\mathbb{1}(x = x') = \begin{cases} 1, & x = x' \\ 0, & x \neq x' \end{cases} \tag{14}$$

Here, d_n is the relative distance from each pixel n to its nearest object boundary. X is the corresponding variable represents this relative distance, which is the horizontal axis of Fig. 7a and Fig. 7b. λ_n is the corresponding weight for pixel n. For the distribution on the annotation probability map, the pixel weight is $\lambda_n = \max_c p^{\text{scrib}}_{c,n}$, the max annotation probability value of pixel n. And the weight is $\lambda_n = \max_c p^{\text{seg}}_{c,n}$ for the distribution on the segmentation probability map. This statistical approach is a soft version of the method in Fig. 1 where $\lambda_n \in \{0, 1\}$ and only labeled pixels are counted.

As shown in Fig. 7a, the distribution of annotation probability is basically consistent with the scribble label, but the segmentation probability is very different from the scribble label. Therefore, using the annotation probability map in the BLP loss to reveal label preference instead of the segmentation probability map is appropriate.

Besides, the statistic of pixel-wise precision in Fig. 7b illustrates the effectiveness of our method for tackling annotation bias. For the baseline model, the pixels close to the object boundary have lower precision than pixels in the central region of the object. It is reasonable that pixels close to the boundary are rarely labeled by scribble, as shown in Fig. 7a. Fortunately, this learning imbalance is alleviated by our BLPSeg, which exceeds the baseline when the relative distance is smaller than 0.2. It means our BLPSeg focuses more on rare labels and consequently improves segmentation accuracy around boundary.

C. BLP Loss

In the above, the BLP loss established on the annotation probability map has been verified to be effective. Here, we further investigate how BLP loss works in network training.

TABLE II

Comparison of Model (w/ LAM) Performance Trained by Various Losses. **CE**: Cross Entropy. **FL**: Focal Loss. **BL**: BLP Loss. All the Models Are Evaluated on PASCAL VOC 2012 *val* Set Without CRF

Loss	mIoU(%)				
	1	2	3	mean	std
CE	73.798	74.004	74.242	74.015	0.222
FL	72.716	72.708	73.339	72.921	0.362
BL	75.604	75.663	75.659	**75.642**	0.033

1) Losses Comparison: As formulated in Eq. 9, BLP loss can be regarded as an extension of focal loss [60] which is first introduced in object detection for hard examples mining. However, scribble-supervised semantic segmentation has a different regime than fully supervised object detection. Therefore, we conduct additional experiments to compare various losses with scribble supervision. As given in Tab. II, simply introducing focal loss even impairs the model performance compared to baseline with cross entropy loss. We believe the reason is that only a small bunch of hard example pixels contribute to the model training by focal loss, which leads to a more severe annotation shortage. In this view, our BLP loss only suppresses the weight of easy example pixels located in the frequently labeled region, leaving more pixel-level labels to serve for network training. As shown in Tab. II, BLP loss brings 1.627% ↑ mIoU improvement, meaning that it is better adapted to the network training with scribble supervision.

2) Loss Map Visualization: Fig. 8 gives a more intuitive perspective on how our BLP loss works. The first row of Fig. 8 provides the images and scribble annotations. And the second row of Fig. 8 shows the predicted annotation probability maps, which have more compact activation regions than the predicted segmentation probability maps in the third row. Because the self-training strategy is introduced in the later period of network training and the scribble is too thin to visualize well, we post the BLP loss map supervised by pixel-level pseudo labels in the fourth row of Fig. 8. It illustrates that BLP loss

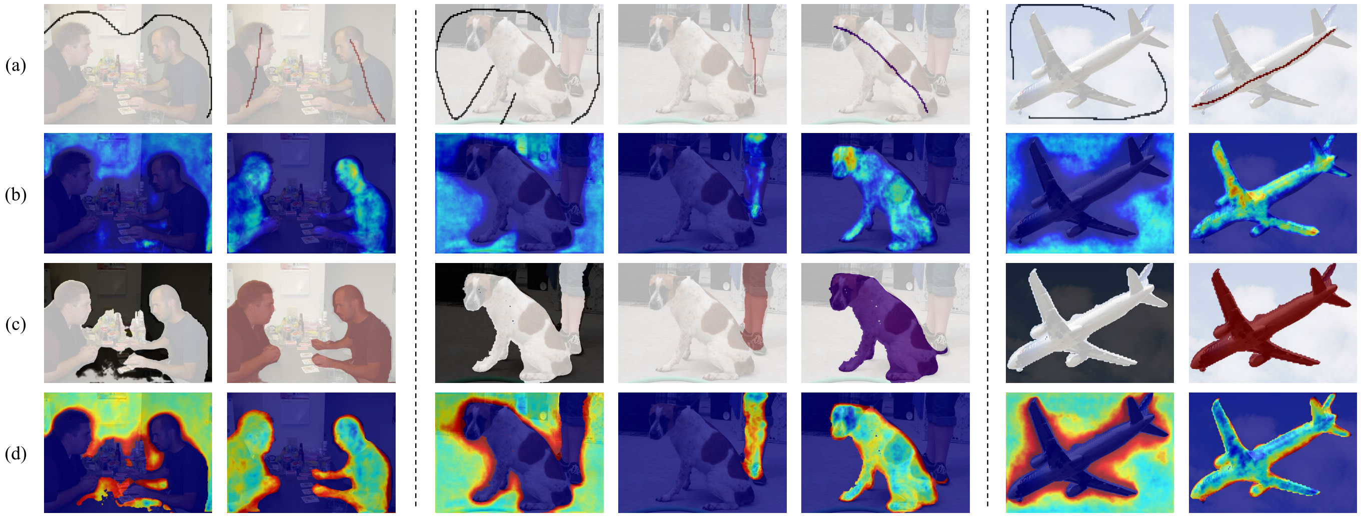


Fig. 8. Visualization for the BLP loss with self-training strategy. (a) Images and scribble. (b) Annotation probability p^{scrib}. (c) Prediction of segmentation p^{seg}. (d) BLP loss map. The BLP loss with the self-training strategy mainly focuses on the boundary region. Best viewed in color.

TABLE III

Experiments for the Similarity Matrix of LAM ($w_{\varnothing} = 0.02$, $\gamma = 2.0$, $itr_{\text{SELF}} = 10k$) on PASCAL VOC 2012 *val* Set Without CRF

Low-level	High-level	Distance	mIoU(%)				
			1	2	3	mean	std
-	-	-	73.302	72.962	73.355	73.206	0.213
✓	-	-	73.482	73.797	73.383	73.554	0.216
-	-	✓	64.555	64.411	64.349	64.438	0.106
✓	-	✓	75.604	75.663	75.659	**75.642**	0.033
-	✓	-	56.323	55.029	56.781	56.044	0.909
-	✓	✓	53.436	52.167	57.284	54.297	2.665
✓	✓	✓	73.493	73.249	73.098	73.280	0.199

with self-training strategy mainly focuses on the regions close to the boundary (colored in red), which is also the rarely labeled region according to Fig. 7a.

D. Local Aggregation Module

1) Feature Aggregation Matrix: The LAM leverages low-level feature similarity and position similarity to build feature aggreagation matrix $\mathbf{A}$ in Eq. 4. We conduct experiments to prove that both are necessary for the LAM. As shown in Tab. III, the model achieves a 0.348% ↑ slight improvement compared to the baseline without LAM when only low-level features are preserved. If the LAM propagates gradient according to the position distance only, the performance decreases to the 64.438% mIoU with a significant 8.768% ↓ drop. And the integration of low-level feature and position similarities in LAM brings a significant 2.436% ↑ improvement compared to the baseline, revealing that both are crucial for supervision propagation.

Besides, we also conducted experiments to construct a feature aggregation matrix using the high-level features of the model. However, the results presented in Tab. III indicate that the utilization of high-level features leads to significant performance degradation, regardless of low-level feature similarity or position similarity. We hold the view that high-level features do not contain enough semantic information in the early stages of network training. Consequently, incorporating high-level feature similarity in LAM at this stage could introduce severe noise during supervision diffusion. In contrast, low-level features contain more stable color information, which is beneficial for supervision diffusion, particularly in the early stages of training. Therefore, the combination of low-level feature similarity and position similarity is the optimal choice for LAM to consistently propagate supervision throughout the entire training process.

TABLE IV

Experiments for the Position Hyperparameter of LAM ($w_{\varnothing} = 0.02$, $\gamma = 2.0$, $itr_{\text{SELF}} = 10k$)

Methods	mIoU(%)				
	1	2	3	mean	std
$\sigma = 3$	75.127	75.387	74.782	75.099	0.303
$\sigma = 6$	75.604	75.663	75.659	**75.642**	0.033
$\sigma = 12$	75.587	74.540	75.219	75.115	0.531
$\sigma = 24$	74.698	73.747	74.697	74.381	0.549
$\sigma = 48$	74.385	73.423	74.323	74.044	0.538
learnable σ	75.533	75.877	75.414	75.608	0.179

Since the hyperparameter σ directly controls the distance regularization range in LAM, we conducted additional experiments to analyze the effect of different choices of σ on the model's performance. As shown in Tab. IV, the performance

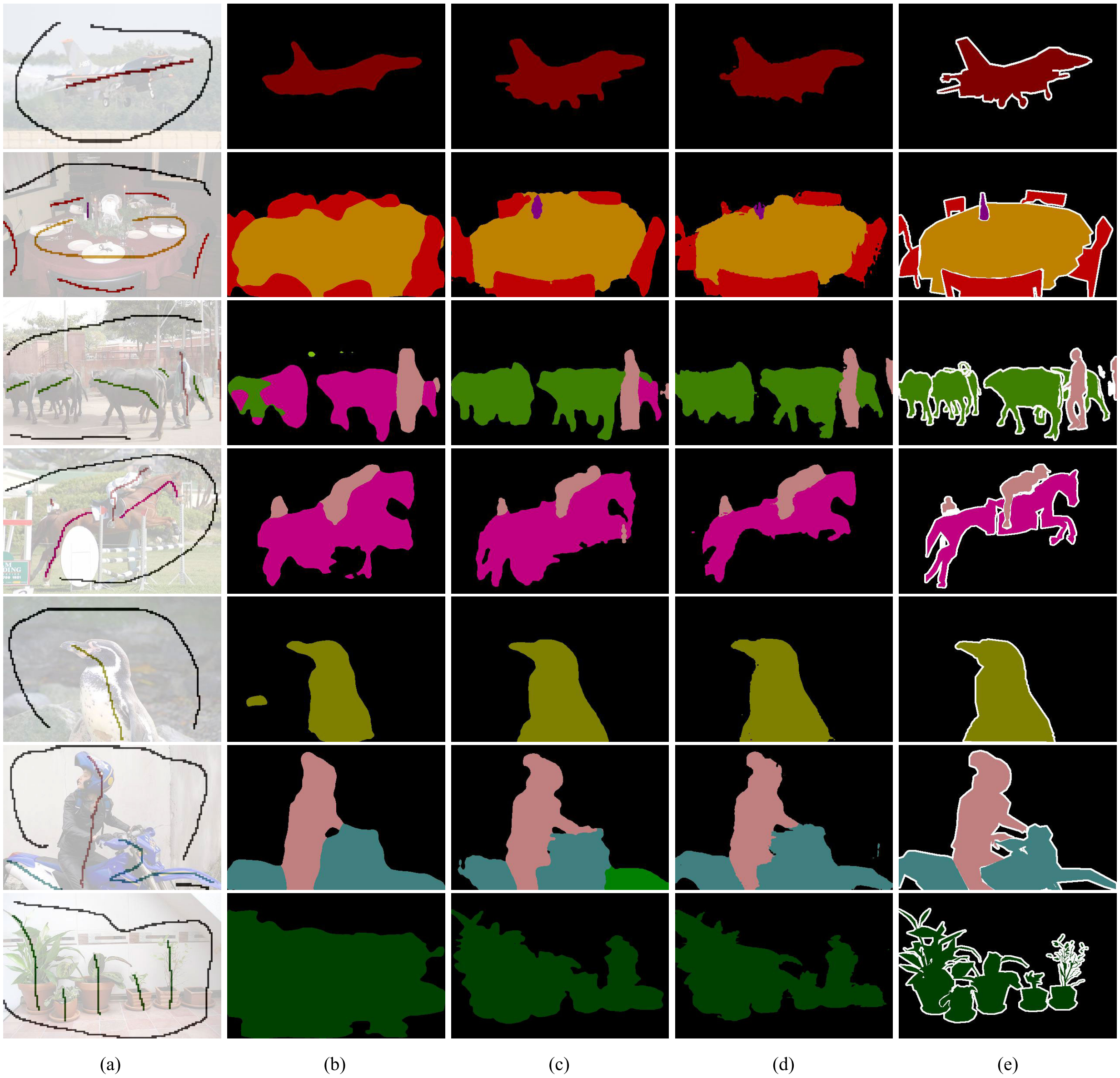


Fig. 9. Visualization of final prediction on PASCAL VOC 2012. (a) Images and corresponding scribble labels. (b) KernelCut [16] predictions. (c) SPML [40] predictions (w/o CRF). (d) BLPSeg predictions (w/o CRF). (e) Ground truth segmentation label.

of BLPSeg does not vary significantly when different static σ values are used. Furthermore, we performed an experiment to explore the potential benefits of using a learnable σ. The results in Tab. IV demonstrate that the model's performance is comparable to the best result achieved with a fixed σ, with only a marginal difference of 0.034% on average. It is worth noting that the learnable σ converged to a value of 4.8, which is relatively close to the model's initial setting of $\sigma = 6$.

2) Aggregation Region: We also specify the aggregation region in LAM to reveal the source of performance improvement. In Tab. V, we separate the features map into labeled and unlabeled regions and conduct an ablation study on them. Compared to the vanilla LAM, which aggregates features from both labeled and unlabeled regions, the LAM variants have lower mIoU. Specifically, the LAM aggregated on labeled region impairs performance by 0.804% ↓ mIoU, and the LAM aggregated unlabeled region only degenerates by 0.065% ↓ mIoU. It shows that the aggregation of unlabeled pixel features is more crucial for LAM. And it is consistent with the original intention of the LAM design that propagates supervision into more unlabeled pixels.

TABLE V
ABLATION STUDY OF LAM AGGREGATION REGION. $\mathcal{L}$: LABELED REGION; $\mathcal{U}$: UNLABELED REGION

$\mathcal{L}$	$\mathcal{U}$	mIoU(%) 1	2	3	mean	std
✓	-	74.652	75.256	74.605	74.838	0.363
-	✓	75.411	75.320	76.001	75.577	0.370
✓	✓	75.604	75.663	75.659	**75.642**	0.033

3) Diffusion Strategy Comparison: Although LAM diffuses supervision in its backward propagation process, it still raises concerns about the difference between the gradient diffusion of

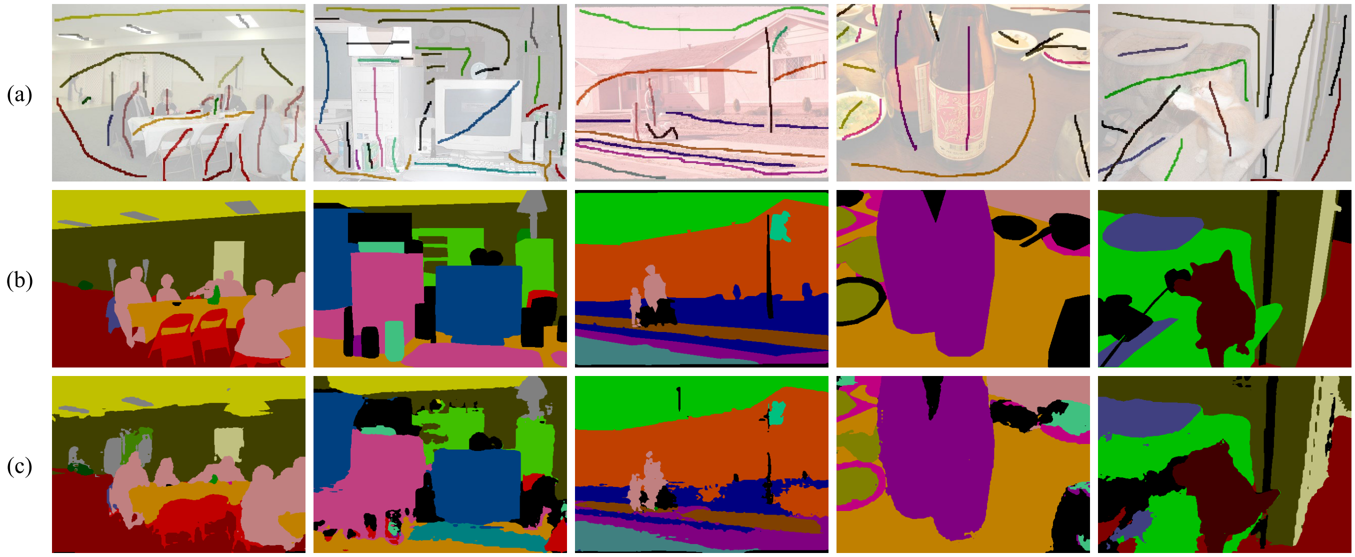


Fig. 10. Visualization of final prediction on PASCAL-Context. (a) Images and corresponding scribble labels. (b) Ground truth segmentation mask label. (c) BLPSeg predictions (w/o CRF).

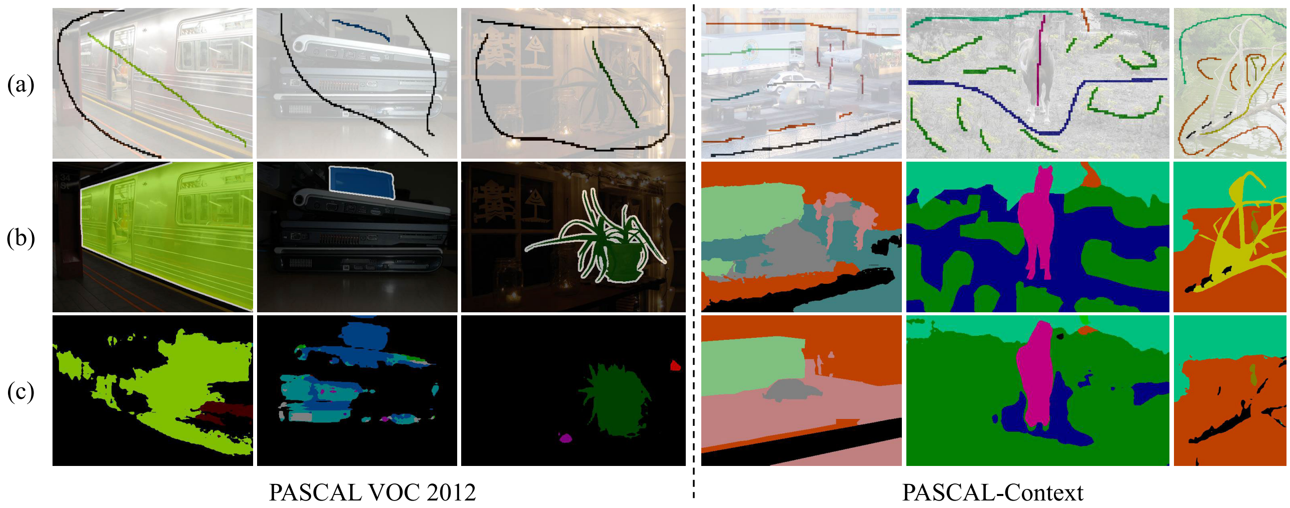


Fig. 11. Visualization of some predicted failure cases of BLPSeg. (a) Images and corresponding scribble labels. (b) Ground truth segmentation mask label. (c) BLPSeg predictions (w/o CRF).

LAM and the classical label diffusion strategy, which expands scribble labels into the larger region and then serves as network supervision. The classical label diffusion method utilizes manually designed strategy, *e.g.,* super-pixels, to expand scribble labels. In our LAM, the supervision diffusion is controlled by $\mathbf{A}$ in Eq. 4, which dynamically learns the low-level feature similarity and is jointly optimized with the whole network. It is a more adaptive approach to propagate supervision than classical label diffusion. As given in Tab. VI, both label and gradient diffusion strategies contribute to the performance improvement compared to the baseline model. And the gradient diffusion of LAM outperforms the label diffusion method by 0.885% ↑ mIoU. We also evaluate the combination of both label and gradient diffusion. The result is close to gradient diffusion only with a slight degradation by 0.055% ↓ mIoU. Considering the simplicity and effectiveness, our BLPSeg abandons classical label diffusion and only retains LAM in practice.

TABLE VI

Comparison of Gradient Diffusion (GD) and Label Diffusion (LD) Strategy (w/o BLP Loss)

GD	LD	mIoU(%) 1	2	3	mean	std
-	-	72.839	72.556	72.528	72.641	0.172
-	✓	73.190	73.669	72.532	73.130	0.571
✓	-	73.798	74.004	74.242	**74.015**	0.222
✓	✓	73.660	74.570	73.650	73.960	0.528

E. Comparison to the State of the Art

1) PASCAL VOC 2012: The performance comparison on PASCAL VOC 2012 validation set is given in Tab. VII. It shows that our BLPSeg achieves state-of-the-art performance compared to other advanced approaches with and

TABLE VII

PERFORMANCE COMPARISON OF OUR BLPSEG WITH OTHER ADVANCED WEAKLY SUPERVISED SEMANTIC SEGMENTATION APPROACHES ON PASCAL VOC 2012. $\mathcal{I}$: IMAGE-LEVEL LABEL, $\mathcal{B}$: BOUNDING BOX LABEL, $\mathcal{S}$: SCRIBBLE LABEL, $\mathcal{F}$: SEGMENTATION LABEL

Method	Publication	Sup.	Backbone	Single-stage	*val* (mIoU%)	
					w/o CRF	w/ CRF
SEAM [37]	CVPR'20	$\mathcal{I}$	ResNet38	-	-	64.5
AdvCAM [68]	CVPR'21	$\mathcal{I}$	ResNet101	-	-	68.1
EDAM [69]	CVPR'21	$\mathcal{I}$	ResNet101	-	-	70.9
CDL [41]	IJCV'23	$\mathcal{I}$	ResNet101	-	71.1	72.4
BCM [9]	CVPR'19	$\mathcal{B}$	ResNet101	-	-	70.2
Box2Seg [11]	ECCV'20	$\mathcal{B}$	ResNet101	-	74.9	76.4
BAP [10]	CVPR'21	$\mathcal{B}$	ResNet101	-	-	74.6
CDL [41]	IJCV'23	$\mathcal{B}$	ResNet101	-	74.4	75.6
ScribbleSup [13]	CVPR'16	$\mathcal{S}$	VGG16	-	-	63.1
RAWKS [14]	CVPR'17	$\mathcal{S}$	ResNet101	✓	59.5	61.4
NormalCut [15]	CVPR'18	$\mathcal{S}$	ResNet101	-	72.8	74.5
GraphNet [70]	ACM MM'18	$\mathcal{S}$	ResNet101	-	70.3	73.0
KernelCut [16]	ECCV'18	$\mathcal{S}$	ResNet101	-	73.0	75.0
BPG [42]	IJCAI'19	$\mathcal{S}$	ResNet101	✓	73.2	76.0
SPML [40]	ICLR'21	$\mathcal{S}$	ResNet101	-	74.2	76.1
PSI [25]	CVPR'21	$\mathcal{S}$	ResNet101	-	74.9	-
A^{2}GNN [71]	TPAMI'21	$\mathcal{S}$	ResNet101	-	-	76.2
PCE [72]	NPL'22	$\mathcal{S}$	ResNet101	✓	72.6	-
CCL [74]	ACM HCMA'22	$\mathcal{S}$	ResNet101	-	74.4	-
TEL [67]	CVPR'22	$\mathcal{S}$	ResNet101	✓	77.1	-
CDL [41]	IJCV'23	$\mathcal{S}$	ResNet101	-	75.2	76.1
AGMM [73]	CVPR'23	$\mathcal{S}$	ResNet101	✓	76.4	-
UperNet [64]	ECCV'18	$\mathcal{F}$	ResNet101	✓	79.3	79.5
BLPSeg	-	$\mathcal{S}$	ResNet101	✓	**77.4**	**77.7**

TABLE VIII

COMPARISON OF OUR BLPSEG WITH OTHER ADVANCED SCRIBBLE-SUPERVISED SEMANTIC SEGMENTATION APPROACHES ON PASCAL-CONTEXT (* DENOTES REIMPLEMENTATED BY OURSELF)

Method	Publication	Supervision	Backbone	Single-stage	*val* (mIoU%)	
					w/o CRF	w/ CRF
ScribbleSup [13]	CVPR'16	$\mathcal{S}$	ResNet101	-	-	36.1
RAWKS [14]	CVPR'17	$\mathcal{S}$	ResNet101	✓	36.0	37.4
GraphNet [70]	ACM MM'18	$\mathcal{S}$	VGG16	-	33.9	40.2
PSI [25]	CVPR'21	$\mathcal{S}$	ResNet101	-	43.1	-
TEL* [67]	CVPR'22	$\mathcal{S}$	ResNet101	✓	45.1	45.1
UperNet [64]	ECCV'18	$\mathcal{F}$	ResNet101	✓	47.1	47.2
BLPSeg	-	$\mathcal{S}$	ResNet101	✓	**45.5**	**45.9**

without post-processed by conditional random field (CRF). The BLPSeg is a single-stage method that is more convenient for implementation in practice, and it outperforms the best existing single-stage method TEL [67] by 0.3% ↑ mIoU without CRF. Moreover, it is noteworthy that our single-stage method surprisingly outperforms various multi-stage methods, and there is only 1.9% mIoU gaps comparing to the fully supervised UperNet baseline without CRF, demonstrating the advance of our work. We visualize some prediction results in Fig. 9 for qualitative comparison.

2) PASCAL-Context: We also evaluate model performance on a more challenging dataset PASCAL-Context with 60 categories. As shown in Tab. VIII, our BLPSeg achieves the highest performance of 45.5% mIoU without CRF. The post-process trick CRF brings another 0.4% performance gain and increases to 45.9% mIoU, which is a new state-of-the-art performance of scribble-supervised semantic segmentation on PASCAL-Context. The performance gap between BLPSeg and its fully supervised upper bound is relatively small, with a difference of only 1.6% mIoU without CRF and 1.3% mIoU with CRF as given in Tab. VIII. Fig. 10 shows some BLPSeg predictions on PASCAL-Context. Because PASCAL-Context contains more categories and the scribble annotation is more complicated, our BLPSeg fails in some details, but it still works well on the most main object and stuff.

3) Failure Cases Discussion: Although our BLPseg achieves a new state-of-the-art performance for scribble-supervised semantic segmentation, there are still some failure cases when browsing through the prediction results. We visualize some of these cases in Fig. 11. In summary, there are four types of scenes where BLPSeg tends to fail: objects with large coverage, objects with unusual perspectives, objects surrounded by extremely complex scenes, and object masks with thin stripes. These cases are also considered challenging for fully supervised semantic segmentation tasks. It is a limitation of BLPSeg that it can only balance label preferences but may struggle to predict accurately on out-of-distribution samples. Addressing these cases may require increasing the dataset size and model scale. And it also presents an interesting challenge for future research.

VI. Conclusion

In this study, we find an annotation bias between scribble labels and object mask labels, *i.e.,* scribble annotations incline to lie in the central area of the object. Therefore, we propose a single-stage scribble-supervised semantic segmentation method BLPSeg to tackle this problem. The BLPSeg first predicts an annotation probability map to reveal the label preference for each image. Then the BLPSeg introduces a novel BLP loss to balance the model training by up-weighting those annotations located in the areas with low annotation probability. The BLPSeg introduces a simple local aggregation module (LAM) to further improve the model training with scribble. The LAM aggregates features by their local similarity and propagates the supervision from labeled pixels to larger unlabeled regions correspondingly during gradient backpropagation. The BLPSeg achieves a promising performance on PASCAL VOC 2012 and PASCAL-Context datasets, proving the effectiveness of it.

Acknowledgment

The code is available at https://github.com/YudeWang/BLPSeg

References

[1] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[2] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.

[3] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6230–6239.

[4] J. Fu et al., "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3141–3149.

[5] T. Wu, S. Tang, R. Zhang, J. Cao, and Y. Zhang, "CGNet: A light-weight context guided network for semantic segmentation," *IEEE Trans. Image Process.*, vol. 30, pp. 1169–1179, 2021.

[6] J. Fu, J. Liu, Y. Wang, J. Zhou, C. Wang, and H. Lu, "Stacked deconvolutional network for semantic segmentation," *IEEE Trans. Image Process.*, early access, Jan. 25, 2019, doi: 10.1109/TIP.2019.2895460.

[7] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.

[8] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele, "Simple does it: Weakly supervised instance and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1665–1674.

[9] C. Song, Y. Huang, W. Ouyang, and L. Wang, "Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3131–3140.

[10] Y. Oh, B. Kim, and B. Ham, "Background-aware pooling and noise-aware loss for weakly-supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 6909–6918.

[11] V. Kulharia, S. Chandra, A. Agrawal, P. Torr, and A. Tyagi, "Box2Seg: Attention weighted loss and discriminative feature learning for weakly supervised segmentation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 290–308.

[12] T. Ma, Q. Wang, H. Zhang, and W. Zuo, "Delving deeper into pixel prior for box-supervised semantic segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 1406–1417, 2022.

[13] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, "ScribbleSup: Scribble-supervised convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3159–3167.

[14] P. Vernaza and M. Chandraker, "Learning random-walk label propagation for weakly-supervised semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2953–2961.

[15] M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers, "Normalized cut loss for weakly-supervised CNN segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1818–1827.

[16] M. Tang, F. Perazzi, A. Djelouah, I. B. Ayed, C. Schroers, and Y. Boykov, "On regularized losses for weakly-supervised CNN segmentation," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2018, pp. 524–540.

[17] B. Zhang, J. Xiao, and Y. Zhao, "Dynamic feature regularized loss for weakly supervised semantic segmentation," 2021, *arXiv:2108.01296*.

[18] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "What's the point: Semantic segmentation with point supervision," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Sep. 2016, pp. 549–565.

[19] X. He, L. Fang, M. Tan, and X. Chen, "Intra- and inter-slice contrastive learning for point supervised OCT fluid segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 1870–1881, 2022.

[20] A. Kolesnikov and C. H. Lampert, "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Sep. 2016, pp. 695–711.

[21] 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 *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6488–6496.

[22] Z. Huang, X. Wang, J. Wang, W. Liu, and J. Wang, "Weakly-supervised semantic segmentation network with deep seeded region growing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7014–7023.

[23] J. Ahn and S. Kwak, "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4981–4990.

[24] C. Redondo-Cabrera, M. Baptista-Ríos, and R. J. López-Sastre, "Learning to exploit the prior network knowledge for weakly supervised semantic segmentation," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3649–3661, Jul. 2019.

[25] J. Xu et al., "Scribble-supervised semantic segmentation inference," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15334–15343.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1–9.

[27] 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," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2016.

[28] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[29] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 603–612.

[30] S. Zheng et al., "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 6877–6886.

[31] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 12077–12090.

[32] A. Dosovitskiy et al., "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[33] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9726–9735.

[34] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 1597–1607.

[35] B. Cheng, A. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 17864–17875.

[36] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Nov. 2020, pp. 213–229.

[37] Y. Wang, J. Zhang, M. Kan, S. Shan, and X. Chen, "Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12272–12281.

[38] X. Li, H. Ma, and X. Luo, "Weaklier supervised semantic segmentation with only one image level annotation per category," *IEEE Trans. Image Process.*, vol. 29, pp. 128–141, 2020.

[39] Y. Xu and P. Ghamisi, "Consistency-regularized region-growing network for semantic segmentation of urban scenes with point-level annotations," *IEEE Trans. Image Process.*, vol. 31, pp. 5038–5051, 2022.

[40] T.-W. Ke, J.-J. Hwang, and S. X. Yu, "Universal weakly supervised segmentation by pixel-to-segment contrastive learning," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–14.

[41] B. Zhang, J. Xiao, Y. Wei, and Y. Zhao, "Credible dual-expert learning for weakly supervised semantic segmentation," *Int. J. Comput. Vis.*, vol. 131, pp. 1892–1908, Apr. 2023.

[42] B. Wang et al., "Boundary perception guidance: A scribble-supervised semantic segmentation approach," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 3663–3669.

[43] A. Obukhov, S. Georgoulis, D. Dai, and L. Van Gool, "Gated CRF loss for weakly supervised semantic image segmentation," pp. 1–14, 2019, *arXiv:1906.04651*.

[44] Z. Pan, P. Jiang, Y. Wang, C. Tu, and A. G. Cohn, "Scribble-supervised semantic segmentation by uncertainty reduction on neural representation and self-supervision on neural eigenspace," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7396–7405.

[45] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.

[46] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[47] A. Vedaldi and S. Soatto, "Quick shift and kernel methods for mode seeking," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2008, pp. 705–718.

[48] K. Haris, S. N. Efstratiadis, N. Maglaveras, and A. K. Katsaggelos, "Hybrid image segmentation using watersheds and fast region merging," *IEEE Trans. Image Process.*, vol. 7, no. 12, pp. 1684–1699, Dec. 1998.

[49] A. Banerjee, I. S. Dhillon, J. Ghosh, S. Sra, and G. Ridgeway, "Clustering on the unit hypersphere using von Mises–Fisher distributions," *J. Mach. Learn. Res.*, vol. 6, no. 9, pp. 1345–1382, 2005.

[50] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[51] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-means clustering algorithm," *J. Roy. Stat. Soc. Ser. C, Appl. Statist.*, vol. 28, no. 1, pp. 100–108, Jan. 1979.

[52] F. R. Chung, *Spectral Graph Theory*, vol. 92. Providence, RI, USA: American Mathematical Soc., 1997.

[53] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[54] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut: Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.

[55] X. Ji, A. Vedaldi, and J. Henriques, "Invariant information clustering for unsupervised image classification and segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9864–9873.

[56] M. Caron et al., "Emerging properties in self-supervised vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9630–9640.

[57] Y. Ouali, C. Hudelot, and M. Tami, "Autoregressive unsupervised image segmentation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Nov. 2020, pp. 142–158.

[58] S. E. Mirsadeghi, A. Royat, and H. Rezatofighi, "Unsupervised image segmentation by mutual information maximization and adversarial regularization," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 6931–6938, Oct. 2021.

[59] R. Harb and P. Knöbelreiter, "InfoSeg: Unsupervised semantic image segmentation with mutual information maximization," in *Proc. DAGM German Conf. Pattern Recognit.* Cham, Switzerland: Springer, Jan. 2021, pp. 18–32.

[60] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.

[61] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2015.

[62] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[64] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun, "Unified perceptual parsing for scene understanding," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 418–434.

[65] J. Lei Ba, J. Ryan Kiros, and G. E. Hinton, "Layer normalization," 2016, *arXiv:1607.06450*.

[66] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.

[67] Z. Liang, T. Wang, X. Zhang, J. Sun, and J. Shen, "Tree energy loss: Towards sparsely annotated semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16886–16895.

[68] J. Lee, E. Kim, and S. Yoon, "Anti-adversarially manipulated attributions for weakly and semi-supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4070–4078.

[69] T. Wu et al., "Embedded discriminative attention mechanism for weakly supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16760–16769.

[70] M. Pu, Y. Huang, Q. Guan, and Q. Zou, "GraphNet: Learning image pseudo annotations for weakly-supervised semantic segmentation," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 483–491.

[71] B. Zhang, J. Xiao, J. Jiao, Y. Wei, and Y. Zhao, "Affinity attention graph neural network for weakly supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8082–8096, Nov. 2022.

[72] M. Li, D. Chen, and S. Liu, "Weakly supervised segmentation loss based on graph cuts and superpixel algorithm," *Neural Process. Lett.*, vol. 54, pp. 2339–2362, Jan. 2022.

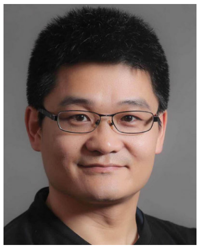
[73] L. Wu et al., "Sparsely annotated semantic segmentation with adaptive Gaussian mixtures," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2023, pp. 15454–15464.

[74] B. Wang, Y. Qiao, D. Lin, S. D. H. Yang, and W. Li, "Cycle-consistent learning for weakly supervised semantic segmentation," in *Proc. 3rd Int. Workshop Hum.-Centric Multimedia Anal.*, 2022, pp. 7–13.

Yude Wang (Student Member, IEEE) received the B.S. degree in computer science from Shandong University in 2017. He is currently pursuing the Ph.D. degree with the Institute of Computing Technology, Chinese Academy of Sciences (CAS). His research interests include semantic segmentation, weakly supervised learning, and semi-supervised learning.

Jie Zhang (Member, IEEE) received the Ph.D. degree from the University of Chinese Academy of Sciences, Beijing, China. He is currently an Associate Professor with the Institute of Computing Technology, Chinese Academy of Sciences (CAS). His research interests include computer vision, pattern recognition, machine learning, particularly include face recognition, image segmentation, weakly/semi-supervised learning, and domain generalization.

Meina Kan (Member, IEEE) received the Ph.D. degree from the University of Chinese Academy of Sciences, Beijing, China. She is currently an Associate Professor with the Institute of Computing Technology, Chinese Academy of Sciences (CAS). Her research interests include computer vision especially face recognition, transfer learning, weakly supervised learning, and deep learning.



Shiguang Shan (Fellow, IEEE) received the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2004. Since 2010, he has been a Full Professor with ICT, CAS, where he is currently the Director of the Key Laboratory of Intelligent Information Processing. His research interests include computer vision, pattern recognition, and machine learning. He has published more than 300 articles in related areas. He was a recipient of the China's State Natural Science Award in 2015 and the China's State S&T Progress Award in 2005 for his research work. He served as an Area Chair for many international conferences, including CVPR, ICCV, AAAI, IJCAI, ACCV, ICPR, and FG. He was/is an Associate Editor of several journals, including IEEE TRANSACTIONS ON IMAGE PROCESSING, *Neurocomputing*, *CVIU*, and *PRL*.

Xilin Chen (Fellow, IEEE) is currently a Professor with the Institute of Computing Technology, Chinese Academy of Sciences (CAS). He has authored one book and more than 400 papers in refereed journals and proceedings in the areas of computer vision, pattern recognition, image processing, and multimodal interfaces. He is a fellow of the ACM, IAPR, and CCF. He is also an Information Sciences Editorial Board Member of *Fundamental Research*, an Editorial Board Member of *Research*, a Senior Editor of the *Journal of Visual Communication and Image Representation*, and an Associate Editor-in-Chief of the *Chinese Journal of Computers* and *Chinese Journal of Pattern Recognition and Artificial Intelligence*. He served as an Organizing Committee Member for multiple conferences, including the General Co-Chair for FG 2013/FG 2018 and VCIP 2022, the Program Co-Chair for ICMI 2010/FG 2024, and an Area Chair/Senior PC for ICCV/CVPR/ECCV/NeurIPS/ICMI for more than ten times.