Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Frozen is better than learning: A new design of prototype-based classifier for semantic segmentation

Jialei Chen [a,*], Daisuke Deguchi [a], Chenkai Zhang [a], Xu Zheng [b], Hiroshi Murase [a]

[a] Nagoya University, Furo-cho, Chikusa-ku, Nagoya, Aichi, Japan

[b] The Hong Kong University of Science and Technology (Guangzhou), No. 1 Du Xue Rd, Guangzhou, Guangdong, China

ARTICLE INFO

Keywords:
Frozen prototype
Contrastive learning
Representation learning
Semantic segmentation

ABSTRACT

Semantic segmentation models comprise an encoder to extract features and a classifier for prediction. However, the learning of the classifier suffers from the *ambiguity* which is caused by two factors: (1) the weights of a classifier for similar categories may have positive similarities lowing the performance for similar categories, named **correlation ambiguity**, and (2) the classifier is prone to predict the category with a larger ℓ_2 norm and vice versa, termed **prior ambiguity**. To comedy the issues, we propose Category-Basis Prototype (CBP), **frozen** and **mutually orthogonalized** prototypes with **equal** ℓ_2 **norm**. Orthogonalization prevents the prototypes from being similar to each other and the equality decouples the prediction from the ℓ_2 norm. To better shape the feature space, we propose Online Centroid Contrastive Loss (OCCL) equipped with centroid and category-level losses. Experiments show that our method yields compelling results over two widely applied benchmarks indicating the effectiveness of our methods.

1. Introduction

As one of the most significant computer vision tasks, semantic segmentation aims to assign each pixel its corresponding category. Benefiting from the development of deep learning, *e.g.*, from ResNet [1] to ViT [2], we have witnessed the great advance of semantic segmentation these years.

To obtain good performance, careful designs for both the encoder to extract features from the input images and a classifier to map the features from the encoder into the category space are significant. For the encoder, as the first fully convolutional network, FCN [3] paves the way for the following segmentation networks. Since then the encoder network has developed from a CNN-based model [3] to a transformer-based model [4].

In addition, for the classifier, besides the original design which is learned by gradient descent algorithm, *i.e.*, 1 × 1 convolution layer, prototype-based classifiers [5], as shown in Fig. 1, have also been achievable recently. Prototype-based classifiers, especially multi-prototype-based classifiers, for a specified category, *e.g.*, a person, the network first separates the area belonging to the category into several pieces by different prototypes. Then the pixels belonging to the corresponding area are forced to be close to the prototypes. Concretely, groups of prototypes are randomly initialized for each category. Then, based on an assignment algorithm, *e.g.*, Sinkhorn–Knopp iteration [5], the pixel-level features are assigned to these prototypes. Finally, these prototypes are updated by the Exponential Moving Average (EMA) [6] based on the assigned features. Moreover, to further expand the inter-class variance, these methods also apply contrastive learning [6] to pull the features and corresponding prototypes together [5] and push others apart.

However, we argue that the learning of classifiers remains sub-optimal attributing to the *ambiguity* which makes the pixels hard to distinguish compared with the ideal situation where all the categories are equal to be classified as shown in Fig. 2(a). The *ambiguity* is caused by: (1) the weights for similar categories, *e.g.*, sofa and chair, obtain positive similarities after training, namely **correlation ambiguity** as shown in Fig. 2(b) and (2) the ℓ_2 norm of each weight is coupled with prediction of one pixel, namely **prior ambiguity** as shown in Fig. 2(c). Correlation ambiguity leads to misclassification among the categories with similar semantics, and prior ambiguity makes the categories with higher ℓ_2 norm easily gain greater confidence and vice versa leading to bias during inference. Meanwhile, the pixel-wise supervision is also insufficient as the intricate semantic information.

To tackle the aforementioned problems, we model semantic segmentation as an implementation of the Expectation Maximization (EM) algorithm by proposing Category-Basis Prototypes (CBP). Formally, we predetermine a group of **mutually orthogonalized** vectors as

[*] Corresponding author.
E-mail addresses: chen.jialei.s6@s.mail.nagoya-u.ac.jp (J. Chen), ddeguchi@nagoya-u.jp (D. Deguchi), zhang.chenkai.d4@s.mail.nagoya-u.ac.jp (C. Zhang), xzheng287@connect.hkust-gz.edu.cn (X. Zheng), murase@nagoya-u.jp (H. Murase).

https://doi.org/10.1016/j.patcog.2024.110431
Received 11 October 2023; Received in revised form 7 February 2024; Accepted 12 March 2024
Available online 21 March 2024

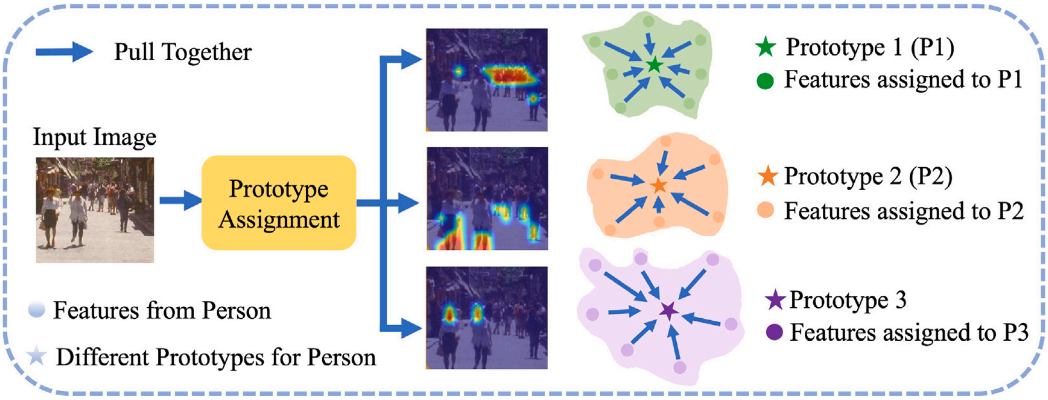
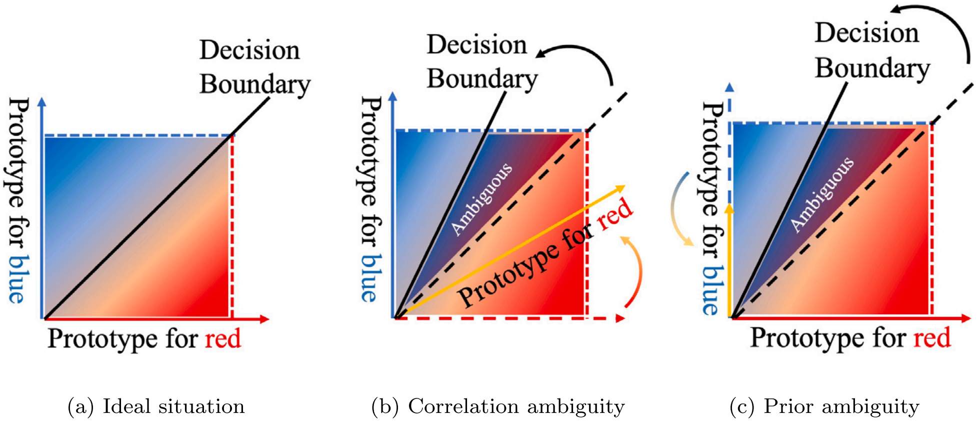


Fig. 1. Illustration of the prototype-based classifier. In this example, there are three prototypes for person, and the prototypes update based on the assigned features.



(a) Ideal situation

(b) Correlation ambiguity

(c) Prior ambiguity

Fig. 2. Different situations of decision boundary: (a) ideal situation, (b) affected by correlation ambiguity, (c) affected by prior ambiguity.

the prototypes for each category and freeze them during training. Consequently, CBP aligns with the pixel-level supervision signal, *i.e.*, one-hot labels, which effectively alleviates the **correlation ambiguity**. Furthermore, we equalize the ℓ_2 norm for each prototype to decouple the prediction from the ℓ_2 norm of the corresponding prototype, which is different from most of the existing works that simply normalize the prototypes [7] to alleviate the **prior ambiguity**. As a result, the semantic segmentation can be viewed as the EM algorithm where the initialization of CBP can be viewed as the E-step, and the training can be modeled as the M-step. However, only relying on the CBP may lead to the relaxation of representations as the decrease in the depth of the network. As a result, we propose the Online Centroid Contrastive Loss (OCCL) module in light of some recent research [6]. OCCL comprises two individual losses: centroid loss and category loss. We assume that different objects belonging to the same category are data augmentation from each other. Based on this assumption, in the former centroid loss, the centroid loss separates the features belonging to the same categories into two groups, serving as two views of one category. Then the view centroids (VC) are produced by averaging the features in each view, and the category centroids (CC) are the average of both views. The InfoNCE loss [8] is then calculated among VC and CC to employ the two VCs of the same category close and apart from other CCs. The latter category loss aims to make each CC apart from other ones and close to its corresponding CBP. As a result, inter-class centroids are apart while intra-class representations are gathering.

Our contributions can be summarized as follows: **(I)** We propose CBP, a new design of classifier, which converts the semantic segmentation to an implementation of the EM algorithm. **(II)** We approach OCCL to better shape the feature space based on CBP. **(III)** Extensive experiments are conducted on two semantic segmentation benchmarks, *i.e.*, ADE20K [9] and COCO-Stuff [10]. Experimental results show that our approach achieves promising results on both datasets and proves that the CBP, *i.e.*, frozen prototype, performs better than either learnable prototypes or classifiers.

2. Related works

2.1. Semantic segmentation

As the first end-to-end semantic segmentation network, FCN [3] applies the model which performs well in the image recognition task, *e.g.*, ResNet [1] as the backbone to extract features and proposes its carefully designed encoder for dense prediction. Since it was proposed, many great works focusing on the design of encoders have improved the ability to extract strong features. For example, DeepLabV3 [11] focuses on enlarging the receptive field, and [12] focuses on utilizing the attention mechanism to extract more representative features. After ViT [2] proves the potential of transformer in computer vision, the transformer-based encoder for semantic segmentation has also been facilitated, and many impressive works, *e.g.*, Segformer [4], have been proposed, and achieve state-of-the-art performance. Recently, with the development of large-scale models, the design of the encoder has also entered a new era. SAM [13] as one of the most representative works, has learned a general notion of what objects are, resulting in a significant impact on computer vision. However, SAM still suffers from a lack of semantic information. To solve these issues, Semantic-SAM [14] are proposed, respectively. Additionally, SAM has also been applied in 3D vision tasks and achieves impressive performance, *e.g.*, anything3d [15] and SAM3D [16]. However, training such models needs lots of data and proper fine-tuning.

Though many works have been proposed for improving the ability of encoders, the design of classifiers, *i.e.*, 1 × 1 convolution layer, remains unchanged for a long time. Recently, some works that apply clustering methods, *e.g.*, ProtoSeg, [5], and generative methods, *e.g.*, GMMSeg [17], have been approached, which can better handle the structure of the training data. However, these classifiers still suffer from correlation ambiguity and prior ambiguity problems. Though several works try to solve the prior ambiguity by normalizing the prototypes [7], the **prior ambiguity** is still ignored. Despite some works, *e.g.*, slimmable dataset [18], applying orthogonality as a regularization term, *i.e.*, forcing the cosine similarity between one feature and others to zero, *e.g.*, DD [19], the prior ambiguity still exists. In this paper, we tackle the prior ambiguity and correlation ambiguity problem together by designing an orthogonal classifier and take a step further to equalize the ℓ_2 norm of the prototypes rather than simply normalizing. Meanwhile, the orthogonality is applied in the initialization of the classifier and is not utilized as a regularization term. Besides, our method can be seen as an assembly between a trainable image encoder and an unlearnable classifier. However, different from training with multi models, *e.g.*, deep reassembly [20], deep graph reprogramming [21], TAM [22], our model contains only one model rather than multi-models. Different from some works that freeze the image encoder during training and train another module to classify novel categories, *e.g.*, CWT [23], PCN [24], and UOTSL [25], our method works in close datasets and freezes classifiers rather than image encoders.

2.2. Contrastive learning

The core idea of contrastive learning is to build positive and negative pairs and enable the positive pairs closer and the negative pairs apart. By mining the mutual information between the positive pairs, the network can learn robust features for vision tasks. Many works, *e.g.*, MoCo [6], have been proposed based on this idea and have transfer ability to downstream tasks. Some works only rely on the positive pairs, *e.g.*, SimSiam [26].

Different from the instance discrimination tasks, the main challenge of segmentation based on contrastive learning is to build positive and negative pairs from dense pixels. The generation can be grouped into online-based methods [5] where the data is sampled from the mini-batch and offline-based methods [7] where the data is sampled from a large memory bank. However, for online-based methods, the generation of the positive pairs is non-stable suffering from the limited batch size. Offline-based methods need lots of memory to store elaborately selected negatives covering all classes. Recently, generative-based methods [17] have been proposed, their data is generated from a distribution but they need to fit a prior ambiguity distribution. In this paper, we propose a novel approach that provides stable prototypes without the need for extra resources such as a memory bank or fitting an ambiguous distribution. Our approach is different from existing works such as NAT [27], where randomly initialized vectors from noise distributions are orthogonalized, and contrastive loss is utilized instead of MSE loss for optimization. Additionally, our approach differs from [5] in that we predefine the prototypes and freeze them during training.

3. Methodology

In this section, we first illustrate the problems that the classifiers in prevalent segmentation models face: the *ambiguity*. To address the issue, we predetermine **mutually orthogonalized** and **frozen** prototypes, *i.e.*, Category-Basis Prototypes (CBP), to convert segmentation into an implementation of EM algorithm based on CBP. To better consider the relationship of pixels, we propose Online Centroid Contrastive Loss (OCCL). OCCL compacts the features belonging to the same category and pushes them from other classes. This section is arranged as follows. Section 3.1 introduces the research question to be resolved in this paper. Section 3.2 introduces the category-basis prototypes. Section 3.3 approaches the OCCL to better shape the feature space. Finally, we introduce training objectives in Section 3.4. The overview of the proposed method can be seen in Fig. 3.

3.1. Problem formulation

Given a set of data $\mathcal{D} = \left\{(x_i, y_i)\right\}_{i=1}^{M}$ where x and y indicate the image and its corresponding label and M denotes the size of the dataset, the goal of semantic segmentation is to assign each pixel its corresponding category. The prevalent design for a semantic segmentation network consists of an encoder $E_\theta(.)$ to extract dense visual features from the input images, and a classifier $g_\phi(.)$ to project the features to the semantic space. The probability that a pixel i is assigned to class l is,

$$p(l|i) = \frac{\exp(g_\phi(f_i)^T \cdot w_l)}{\sum_n \exp(g_\phi(f_i)^T \cdot w_n)}, \tag{1}$$

where $p(l|i) \in [0, 1]$ indicates the probability that pixel i is assigned to class l, $f_i \in \mathbf{F}$ is the feature of pixel i where $\mathbf{F} \in \mathcal{R}^{B*C*H*W}$ indicates the features from $E_\theta(.)$, $n \in N$ denotes the number of categories. $\mathbf{W} \in \mathcal{R}^{N*C}$ indicates the prototypes for all N categories, $w_l \in \mathbf{W}$ implies the lth prototype from $\mathbf{W}$. The inner product in Eq. (1) equals,

$$g_\phi(f_i)^T \cdot w_l = \|g_\phi(f_i)\| \cdot \|w_l\| \cdot \cos(g_\phi(f_i), w_l), \tag{2}$$

where $\|\cdot\|$ denotes the ℓ_2 norm of a vector and $\cos(\cdot,\cdot)$ denotes the cosine similarity between two vectors. Eq. (2) implies that $p(l|i)$ relates to the ℓ_2 norm of the classifier's weights, except for the cosine similarity between features and the weights of the classifier. We summarize this problem as **prior ambiguity**. Meanwhile, we observe that the classifier's weights for the categories with similar semantics always have positive similarities with each other. Though this fact aligns with people's intuition, it violates the pixel-level supervision signal, *i.e.*, one-hot labels, which are orthogonal with each other, failing to preserve the positive similarities between similar categories. This problem is termed **correlation ambiguity**. These two problems impede the generalization ability of the segmentation models, namely *ambiguity*, *i.e.*, a pixel may be misclassified to the classes which have similar semantic information or the classes with more pixels in a dataset. Though research endeavors have made efforts in solving the **prior ambiguity** [7], the **correlation ambiguity** which hinders the perception of correlation between categories with similar semantics is always ignored. Meanwhile, we take a step further to expand the simple normalization to equalization. Differently, we propose to solve the ambiguity by redesigning the classifier, *i.e.*, CBP.

3.2. Category-basis prototype

To address the *ambiguity*, we propose the category-basis prototype. Specifically, suppose there are N unique categories in a dataset. Before the training process, We **randomly** initialize N vectors based on Kaiming Initialization [28] $\mathbf{W} \in \mathcal{R}^{N*C}$ where C denotes the channel number. Then we orthogonalize $\mathbf{W}$ by the Gram–Schmidt algorithm and equalize the ℓ_2 norm of the orthogonalized vectors. The overall initialization process can be shown in Algorithm 1.

For the correlation ambiguity, as we freeze the category-basis prototypes they are always orthogonalized with each other, which prevents them from being similar to each other. For the prior ambiguity, the ℓ_2 norm of each weight is equal, and the classification probability is decoupled from the ℓ_2 norm of both the representation and the prototypes, only depending on the cosine similarity with the prototypes.

Besides, another benefit of applying category-basis prototypes is that the semantic segmentation is converted to a clustering problem based on frozen prototypes. More precisely, the initialization of CBP can be viewed as the E-step in the EM algorithm [26] where the centers of

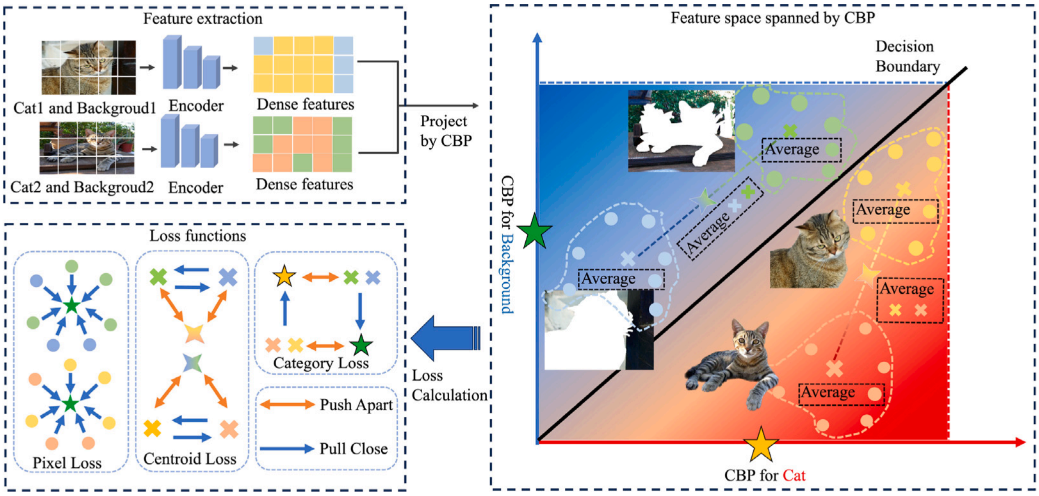


Fig. 3. The illustration of the overall framework. The input images are first fed into an encoder to obtain dense features. Then, these features are cast to the space spanned by the CBP. Finally, the pixel-level loss, *i.e.*, cross-entropy, and OCCL is applied for optimizing.

Algorithm 1 The initialization of category-basis prototypes. We first randomly initialize N vectors from Kaiming initialization [28], where N denotes the category number in a dataset, and the ℓ_2 norm of the vectors T.

```
Randomly initialize $\mathbf{W} \in \mathcal{R}^{N*C}$ # Initialize the prototypes to be orthogonalized.
$\beta = \emptyset$, $\alpha = \emptyset$, $T$ # Initialize the temporal results and the $\ell_2$ norm value.
$\beta$.append($\mathbf{W}[1]$), $\alpha$.append($\mathbf{W}[2], \mathbf{W}[3]...\mathbf{W}[N]$) # Separate the $W$ and assign them to $\alpha$ and $\beta$.
for each $a_i \in \alpha$ do
    $temp = 0$ # Initialize a temporal result.
    for each $b_j \in \beta$ do
        $v = \frac{a_i^T \cdot b_j}{b_j^T \cdot b_j}$ # Compute the weight for each $b_j$.
        $temp = temp + v \cdot b_j$ # Update the temporal result.
    end for
    $\beta$.append($\alpha_i - temp$) # Add the temporal results to the list of $\beta$.
end for
$\beta = \frac{\beta \cdot T}{\sqrt{\beta^T \cdot \beta}}$ # Equalize the $\ell_2$ norm of the prototypes.
return $\beta$
```

each category are defined. However, the CBP is initialized **without any prior knowledge**, *e.g.*, **language information** [29], or **any update**, *e.g.*, gradient decent [3] or EMA [5]. The optimization of the network can be seen as the M-step where each pixel is pulled together with the corresponding centers. Different from the existing works, our method provides a frozen center and extends normalization to equalization. Moreover, CBP spans a linear space with the same dimension, *i.e.*, N, as the supervision signal.

3.3. Online centroid contrastive loss

After mitigating the correlation and prior ambiguity problem at the pixel level, we further propose to measure the relationships among pixels, *e.g.*, intra-class compactness, to improve the segmentation performance. To better shape the feature space, we approach Online Centroid Contrastive Loss (OCCL). First, we define the View Centroid (VC) v_{li} as the average of the features belonging to lth category in the ith view. More precisely, given the features $\mathbf{F} \in \mathcal{R}^{B*C*H*W}$ extracted from $E_\theta(.)$ and their label y, where B, C, H, W denote the batch size, channel, height, and width of the features, respectively. We equally divide y into two parts and get y_1 and y_2 as two different views of y where y_0 contains the same number of categories as y while the number of objects is half. Then v_{li} can be computed as,

$$v_{li} = \frac{\sum_{B,H,W} \mathbf{F}[\mathbb{1}(y_i = l)]}{\sum_{B,H,W}[\mathbb{1}(y_i = l)]}, \tag{3}$$

where $\mathbb{1}$ implies whether the pixel belongs to the category l, and $i \in [1,2]$ indicates the ith view. The OCCL consists of centroid loss and category loss.

Centroid Loss. Motivated by the recent advance in contrastive learning [6], we assume different objects in a mini-batch belonging to the same category as a data augmentation from each other. During training, we split the y into y_1 and y_2. Then, given $\mathbf{F}$ and y_1, y_2, the v_{l1} and v_{l2} are calculated by Eq. (3). Then the centroid for category s_l is obtained by averaging v_{l1} and v_{l2}, namely Category Centroid (CC). The contrastive loss driven by InfoNCE [8] can be calculated between VC and CC,

$$\mathcal{L}_v(v_{lx}) = \Sigma_i^N \frac{\exp(v_{i1}^T \cdot v_{i2}/\tau)}{\Sigma_{j\neq i}^N \exp(v_{jx}^T \cdot SG(s_j)/\tau) + \exp(v_{i1}^T \cdot v_{i2}/\tau)}, \tag{4}$$

where τ is the hyper-parameter to scale the inner product among all the centroids and SG is the stop gradient operation, N indicates the number of categories, and $x \in [1,2]$ indicates the index of view.

Category Loss. Besides the centroid loss, we propose category loss to make each category closer to its corresponding CBP. Given view centroid v_{l1} and $w \in \mathbf{W}$, the category loss is obtained by,

$$\mathcal{L}_c(v_{l1}) = \Sigma_i^N \frac{\exp(v_{i1}^T \cdot w_i/\tau)}{\Sigma_{j\neq i}^N \exp(v_{j1}^T \cdot w_j/\tau) + \exp(v_{i1}^T \cdot w_i/\tau)}, \tag{5}$$

The total loss function of OCCL is,

$$\mathcal{L}_{occ}(\lambda, \phi) = \frac{\lambda * (\mathcal{L}_v(v_{l1}) + \mathcal{L}_v(v_{l2})) + \phi * (\mathcal{L}_c(v_{l1}) + \mathcal{L}_c(v_{l2}))}{2}, \tag{6}$$

where λ and ϕ are two hyperparameters scaling the corresponding loss.

Relations to the previous paradigm. The contrastive loss can be modeled as a matching problem between queries and keys [6]. For centroid loss, our method differs from the existing works in the generation of queries and keys. The queries and keys are all generated in an online manner, while the related work [30] obtains keys from a large memory bank. Consequently, our method is simpler, easier to implement, and more efficient.

For the category loss, similar to the random initialization strategy in NAT [27], we also sample the prototypes randomly from a noise distribution, but the orthogonal operation brings category-related information to the prototypes. In addition, the keys are all frozen, which is different from the existing works that update them by either gradient decent [31] or EMA [5].

Table 1
Quantitative results on ADE20K and COCO-Stuff datasets.

ADE20K					COCO-Stuff				
Pub	Head	Backbone	mIoU	Δ	Pub	Head	BackBone	mIoU	Δ
ECCV18	DeepLabv3+ [11]	RN101 [1]	44.1	–	CVPR19	SVCNet [32]	blue [1]	39.6	–
NeurIPS21	MaskFormer [33]	blueRN101 [1]	46.0	–	ECCV18	DANet [34]	RN101 [1]	39.7	–
ECCV20	OCR [12]	HRFormer-B [35]	48.7	–	ICCV19	SpyGR [36]	RN101 [1]	39.9	–
ICCV21	UPerNet [37]	Swin-B [38]	48.0	–	NeurIPS21	MaskFormer [33]	RN101 [1]	39.8	–
CVPR22	ProtoSeg [5]	Swin-B [38]	48.6	+0.6	CVPR22	ProtoSeg [5]	Swin-B [38]	42.4	+0.9
NIPS22	GMMSeg [17]	Swin-B [38]	49.0	+1.0	NIPS22	GMMSeg [17]	Swin-B [38]	44.3	+0.7
ICLR23	DNC [39]	Swin-B [38]	48.6	+0.6	ICLR23	DNC [39]	Swin-B [38]	–	–
CVPR15	FCN[a] [3]	RN101[a] [1]	39.9	+1.3	CVPR15	FCN[a] [3]	RN101[a] [1]	32.5	+0.5
	FCN [3] + ours		**41.2**			FCN [3] + ours		**33.0**	
CVPR22	UperNet [37][a]	ConvNext-B[a] [40]	48.9	+0.7	CVPR22	UperNet[a] [37]	ConvNext-B[a] [40]	43.6	+0.6
	UperNet [37] + ours		**49.6**			UperNet[a] [37] + ours		**44.2**	
CVPR21	UperNet[a] [37]	Swin-B[a] [38]	47.8	+0.7	CVPR21	UperNet[a] [37]	Swin-B[a] [38]	41.5	+1.1
	UperNet [37] + ours		**48.5**			UperNet [37] + ours		**42.6**	
NeurIPS21	Segformer[a] [4]	MiT-B5[a] [4]	48.9	+1.0	NeurIPS21	Segformer[a] [4]	MiT-B5[a] [4]	43.4	+1.5
	Segformer [4] + ours		**49.9**			Segformer [4] + ours		**44.9**	

[a] Indicates reimplemented method. Δ indicates the improvement compared with the baseline performance.

3.4. Training objectives

Before training, we first initialize the category-basis prototypes based on the algorithm in Section 3.2. Then we employ both the cross-entropy at the pixel level and the OCCL at the centroid level. In a nutshell, the total loss is

$$\mathcal{L} = \mathcal{L}_p(E_\theta(x)^T \cdot W, y) + \mathcal{L}_{occ}(\lambda, \phi), \tag{7}$$

where $\mathcal{L}$ indicates the total loss, $\mathcal{L}_p$ denotes the cross entropy loss at pixel-level between the prediction and its ground truth y, $\mathcal{L}_{occ}(\lambda, \phi)$ indicates the OCCL.

4. Experiments

4.1. Experiment setup

Dataset and Implementation Details. We demonstrate our results on two semantic segmentation benchmarks: ADE20K [9] and COCO-Stuff [10]. ADE20K is a large-scale scene parsing benchmark dataset that covers 150 categories. The dataset contains a training dataset with 20K images, a validation dataset with 2K images, and a test dataset with 3K images. COCO-Stuff contains 10K images including 9K training images and 1K test images. There are 80 object categories, 91 stuff categories, and 1 unlabeled.

Our codes are based on the MMsegmentation, following the default settings for each dataset. Specifically, all the backbones are first pre-trained on ImageNet1K and the rest layers are randomly initialized. The augmentation techniques include random scale jittering with a factor in [0.5, 2], random horizontal flipping, random cropping, and random color jittering. For convolution models the optimizer is SGD and for transformer-based models is AdamW. The learning rate is scheduled following the polynomial annealing policy. In addition, the batch size for both datasets is set to 16. The crop size is set to 512 pixels × 512 pixels. The models are trained for 160K and 80K iterations on ADE20K and COCO-Stuff, respectively. τ is set to 0.07 and the length is set as 2 by default.

Evaluation. For both datasets, we rescale the short scale of the image to train crop size while keeping the aspect ratio unchanged. Our model is trained and tested on 8 V100 (32 GB) GPUs. We report the mean intersection over union (mIoU) score for each model. **Note that we do not use tricks, *e.g.*, multi-scale testing, test-time augmentation, in inference.**

Table 2
Resource consumption comparison and Δ indicates the performance improvement compared with the baseline performance.

Model	Backbone	Memory	fps	Params	GFlops	Δ
Swin [38]	Base [38]	10079MB	23.4	120.0M	300.0	–
FCN [3]	RN101 [1]	18439MB	28.8	66.2M	276.0	–
GMMSeg [17]	MiT-B5 [4]	30513MB	10.2	84.9M	111.0	+0.6
Segformer + Ours		7021MB	19.5	82.1M	75.6	**+1.0**
Segformer	MiT-B4 [4]	5969MB	24.1	61.4M	59.3	+0.5
+Ours		6071MB	23.9	61.5M	59.6	
ConvNext [40]	Base [40]	5279MB	16.6	121.0M	293.0	+0.7
+Ours		6021MB	16.5	121.0M	296.0	

4.2. Comparison with state-of-the-art

ADE20K. Table 1 reports the results of representative segmentation models on ADE20K dataset. For different models, we conduct extensive experiments and choose the hyperparameter that can obtain the best performance. In a specific, λ is set as 0.4 and the ϕ is set to 0.1 for FCN [3], 1, 0.2 for ConvNext [40], Swin Transformer [38], and 1, 0.1 for Segformer [4]. Besides, the length of the CBP is set as 1 for Swin-Transformer [38] and Segformer [4]. Our models gain the expected improvements on both convolutional-based and transformer-based models. In specific, for convolutional models, combining our proposed CBP and OCCL outperforms the original baseline by a large margin which is **1.3%** and **0.7%** mIoU for FCN and ConvNext, respectively. For the transformer-based models, *i.e.*, Swin-transformer and Segformer, applying our CBP and OCCL achieves significant improvements of **0.7%** and **1.0%** mIoU than the baseline performance. The qualitative results on the ADE20K dataset are shown in Fig. 7. Benefiting from the proposed CBP and OCCL, the model can outperform its corresponding baseline methods. Meanwhile, compared with the SOTA methods on the performance improvements, we can find that under the same backbone, *i.e.*, SwinTransformer-Base [38], our methods can achieve similar improvements to ProtoSeg [5] and DNC [39], and a little bit lower improvement than GMMSeg [17]. However, our methods need fewer resources than GMMSeg.

We further analyze the consumption of computational resources for the proposed methods as shown in Table 2. Note that in this experiment, we use one RTX A6000 GPU to report the performance. Compared with the SOTA methods, *e.g.*, GMMSeg [17], under the same Segformer-B5 backbone, our methods can achieve similar mIoU scores on the ADE20K dataset while costing less $\frac{1}{4}$ GPU memories (7 GB vs. 30 GB) during training. Besides, during testing, the proposed methods

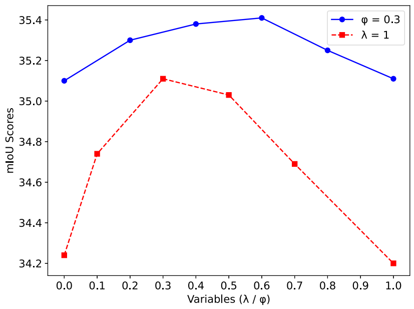


Fig. 4. Ablations on training objectives. X-axis shows the non-frozen parameter. When freezing λ, the x-axis implies ϕ and vice versa. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Ablation on the effect of CBP and OCCL.

CBP	OCCL	mIoU	mAcc
–	–	33.4	44.4
✓	–	33.5	44.4
–	✓	34.3	45.8
✓	✓	**35.1**	**48.1**

Table 4
Ablation on update strategies of CBP.

Update	mIoU	mAcc
Frozen	**35.1**	**48.1**
Gradient	34.6	46.1
EMA	29.9	41.8
Penalty	34.4	45.9

can achieve over twice as fast inference speed (19.5 fps vs. 10.2 fps) and much fewer GFlops (75.6G vs. 111.0G) and number of parameters (82.1M vs. 84.9M). Meanwhile, we can achieve higher improvements (1.0 vs. 0.6) compared with GMMSeg [17]. Compared with the baseline methods, our methods can achieve exceptional improvements, *e.g.*, 0.5% in mIoU using Segformer-B4 backbone with little increase of resources.

COCO-Stuff10K. We also conduct experiments on the COCO-Stuff dataset to demonstrate the effectiveness of our proposed CBP and OCCL as shown in Table 1. For different models, the hyperparameters are different. In a specific, λ is set as 1 and the ϕ is set to 0.3 for FCN [3], ConvNext [40], Swin Transformer [38], and 1, 0.1 for Segformer [4]. Besides, the length of the CBP is set as 0.3 for FCN [3], 3 for ConvNext [40], and 1 for Segformer [4]. For convolutional-based models, compared with the basic models without our methods, the performance gain is **0.5%** and **1.1%** in FCN and ConvNext, respectively. Our methods consistently outperform the baseline of transformer-based models. More precisely, CBP and OCCL bring performance gains of **0.6%** and **1.5%** mIoU with Swin-transformer and Segformer models. When it comes to performance improvement, we can achieve even higher performance gain than the SOTA methods, *i.e.*, GMMSeg and ProtoSeg, which indicates the effectiveness of our methods.

4.3. Ablation study

To investigate the effectiveness and rationality of our proposed CBP and OCCL, we use the Segformer-B0 model as the baseline and conduct ablation studies on the ADE20K [9] dataset. The model we use in this section is trained with 40K iterations. For the OCCL, the ϕ is set to 0.3, λ as 1.0.

The effectiveness of the proposed modules. We first investigate the effectiveness of the newly proposed methods, *i.e.*, CBP and OCCL, as shown in Table 3. We first ablate all the proposed modules to obtain a base model for evaluating the proposed modules. The baseline model achieves 33.4% mIoU score and 44.4% mAcc scores. Then we only introduce CBP to the baseline model, and we find that this model achieves similar performance which is 33.5% in mIoU and 44.4% in mAcc. We think this is because CBP is discriminative enough as the orthogonality, which proves the effectiveness of the proposed CBP. Then we validate the effectiveness of the OCCL, and we can observe that the performance gain is expected which is 0.9% in mIoU score and 1.4% in mAcc. Finally, we combine the two proposed modules and achieve a performance gain by a large margin which is 1.7% in mIoU score and 3.3% in mAcc. The ablation studies above prove that both of the proposed modules can bring merit to the semantic segmentation task. We also visualize the similarity map of the weights in the classifier for the baseline model, *i.e.*learned by gradient descent, and the proposed ones, *i.e.*, CBP. As there are 150 categories in ADE20K [9] dataset, we only randomly sample some categories as shown in Fig. 6.

Different update strategy for the category-basis prototype. In Section 3.2, we argue that freezing the CBP during the training process is a good choice as it matches the supervision signal. To investigate whether freezing the CBP is an optimal way for the category-basis prototype, we conduct experiments with different update strategies for the category-basis prototype. We apply the same initialization, *i.e.*, orthogonalization, and equalization at the beginning of training, and the only difference among them is how to update the parameters in the CBP. We set our frozen CBP as the baseline. As shown in Table 4, we compare our frozen CBP with the updated CBP by gradient and clustering. In this experiment, “EMA” means that the CBP is updated by the EMA of CC. We can find that no matter whether the CBP is updated with the gradient-based or cluster-based method, the performance of will drop drastically, *i.e.*, from **35.1%** → 29.9% in mIoU and **48.1%** → 41.8% in mAcc for clustering-based methods. The drop is from **35.1%** → 34.6% in mIoU and **48.1%** → 46.1% in mAcc for the gradient-based method, which shows that the frozen prototypes are beneficial for semantic segmentation as preserving the orthogonality. At the same time, we can further prove that a frozen discriminative initialization is better than the learning paradigm. Besides, we further conduct an experiment where the CBP is learnable and always equalized (making the ℓ_2 of prototypes the same) during training, and we add another orthogonality loss with a weight of 0.01, *i.e.*, forcing the cosine similarity between one prototype and others to 0, same as [18] to penalize the prototype. The result is shown in the row of “Penalty” in Table 4. The mIoU is lower than both Frozen and Gradient ones. The reason we think is that penalizing the classifier rather than the feature space leads to a constrained classifier and confused feature space. In contrast, CBP is mutually orthogonalized vectors with the same nature as the one-hot label, *i.e.*, supervision, and enables the feature space more distinguished.

Training Objectives. We first investigate our training objectives, *i.e.*, Eq. (7). As $\mathcal{L}_{occ}$ consists of $\mathcal{L}_v$ and $\mathcal{L}_c$, we ablate these two losses respectively as shown in Fig. 4. To show the effect of both hyperparameters in the same figure, the x-axis represents the variable that is not frozen, *e.g.*, if λ is fixed, the x-axis represents the change of ϕ and vice versa. For $\mathcal{L}_v$, we freeze the hyper-parameter of category contrastive learning, *i.e.*, ϕ, as 0.3, and investigate the effectiveness of centroid contrastive learning as shown in the orange line. Without the centroid loss, the performance drops at most 0.3%, indicating the significance of centroid loss. Meanwhile, the scale of the centroid contrastive learning should also be carefully considered as the performance ranges from **35.4%** to 35.1% which is a large gap. Then we freeze the centroid loss and examine whether the category loss contributes to the model as shown in the blue line. In Fig. 4, the performance of the segmentation model benefits a lot from the category loss as the performance gap between best and worst (without category loss) performance is near **1%**. Without category loss, the performance drops from 35.1% to 34.2%.

Table 5
Ablation on initialization of CBP.

Equal	Ortho	mIoU	mAcc
✓	✓	**35.1**	**48.1**
–	✓	33.3	44.6
✓	–	35.0	47.8
✓	CLIP	33.2	45.2
✓	Masked	30.3	41.7
✓	Masked + ✓	35.1	48.0

Table 6
Ablation study on orthogonalization.

Initialization	Categories	Sum_IoU
Equal+Ortho Equal	bike+minibike	92.3 92.0 (↓ 0.3)
Equal+Ortho Equal	stairs+stairways	53.8 52.8 (↓ 1.0)
Equal+Ortho Equal	oven+microwave	37.4 36.1 (↓ 1.3)

Table 7
Ablation study on equalization.

Initialization	Category	Amounts	mIoU
Ortho+Equal Ortho	flag	105460	24.3 17.0 (↓ 7.3)
Ortho+Equal Ortho	radiator	20652	28.9 23.6 (↓ 5.3)
Ortho+Equal Equal	wall	39565000	68.4 68.4

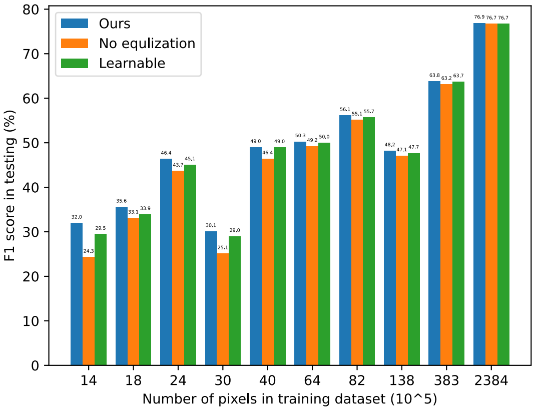


Fig. 5. Quantification of prior ambiguity.

Table 8
Quantification on the correlation ambiguity.

Initialization	CS ↓	mIoU (%) ↑
Equal+Ortho	**0**	**35.1**
Equal	6.2	35.0
Learnable	75.8	34.6
Penalty	73.2	34.4

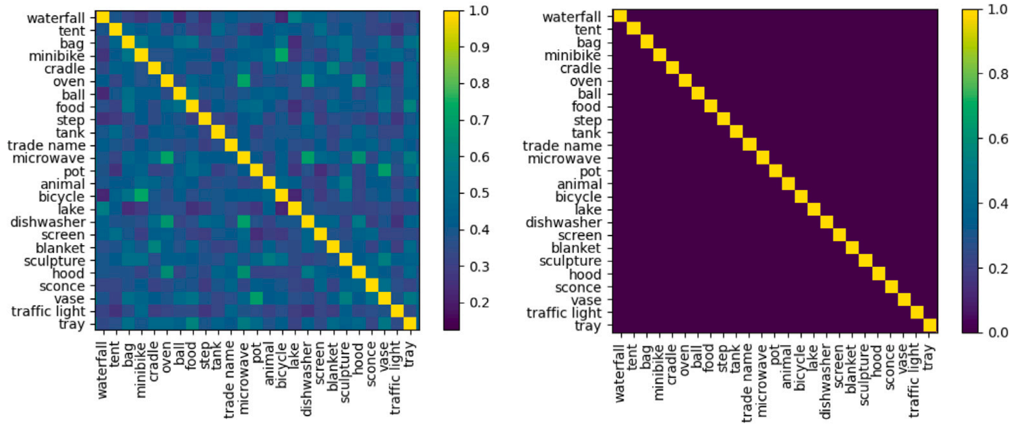
Different initialization of the category-basis prototype. To explore whether the prior ambiguity and correlation ambiguity problem can be mitigated, we yield an ablation experiment to validate the effectiveness of equalization and orthogonalization shown in Table 5 where Equal denotes whether equalization is applied and ortho indicates whether the orthogonalization is applied. By default, we set applying both orthogonalization and equalization to initialize the CBP as a performance baseline. Then we ablate equalization and orthogonalization respectively. The baseline can achieve the best performance in mIoU score and mAcc, *i.e.*, **35.1%** and **48.1%** at the same time. When removing equalization, the performance drops drastically to 33.3% and 44.6%. If the orthogonalization is ablated, though the mIoU score does not change drastically, *i.e.*from 35.1% to 35.0%, the mAcc drops to 47.8%. Moreover, instead of applying the orthogonality, we initialize the CBP with the text features produced by the CLIP [29] ViT [2] text encoder. Compared with the CBP, there is a large performance drop in both mIoU and mAcc scores, *i.e.*, from 35.1% to 33.2% in mIoU and 48.1% to 45.2% in mAcc. The experiments above imply that both initialization, *i.e.*, orthogonality and equalization, contribute to the higher performance of our methods. Even the features obtained from foundation models, *e.g.*, CLIP, may not be proper for the semantic segmentation tasks. Moreover, we use the pretrained backbone of Segformer-B0 to extract category-level features as prototypes. Note that these prototypes are also fixed during training. As can be seen from the second last row of Table 5, performance reaches its worst, *i.e.*, 30.3% in mIoU and 41.7% in mAcc. However, when we orthogonalize them, the performance is very close to the one with CBP, indicating that the orthogonality rather than the initialization has a larger impact on the performance.

Besides, to further investigate if the correlation ambiguity and prior ambiguity are mitigated, we conducted another two experiments. In the first experiment to validate the correlation ambiguity, we randomly select two categories with similar semantic categories, *e.g.*, sofa and chair, and compute their sum mIoU score so that we can test whether **correlation ambiguity** is mitigated. As shown in Table 6, we can observe that without the orthogonalization, the sum mIoU score will drop in different scales. For instance, for bike and minibike, the performance drops **0.3%**, and for stairs and stairways, it drops **1.0%**. Moreover, to quantify the class separability and the prior ambiguity, we apply the confusion matrix to describe the prediction of each pixel and the equations to show class separability and prior ambiguity,

$$CS = 0.5 * \sum_{i}^{N} \sum_{j \neq i}^{N} \frac{w_i^T \cdot w_j}{\|w_i\| * \|w_j\|}, \tag{8}$$

where CS is the class separability $w \in \mathbf{W}$ means the prototypes for classification. A large CS indicates a positive correlation and little distance between the prototypes, *i.e.*, large class separability. The results are shown in Table 8. With both normalization and the orthogonality, the CS and the mIoU achieve their peak, *i.e.*, 0 and 35.1%. When removing the orthogonality ("Equal"), the class separability and the mIoU drop to 6.2 and 35.0%. If the prototype is learnable ("Learnable"), the class separability decreases drastically to 75.8 and 34.6%. For the "Penalty", as the classifier is constrained, though the CS is lower, mIoU is also lower than "Learnable".

To investigate if the prior ambiguity can be tackled by our methods, we conducted another experiment. We select several categories with large and small amounts of pixels, *i.e.*, flag, radiator, and wall, in the ADE20K dataset. We can observe from Table 7 that without equalization, the performance of the categories with fewer pixels may descend by a large margin. For instance, the flag obtains a large performance drop, *i.e.*, **7.3%**, and the radiator **5.3%**. By contrast, the wall that maintains the pixels **100**+ times larger than the previous two categories does not change. This experiment proves the effectiveness of our proposed methods. Moreover, note that the relation factor between the number of pixels in the training dataset and the ℓ_2 norm of the learnable prototypes have a positive relation factor, *i.e.*, 0.27, the prior ambiguity always happens in the categories with fewer pixels. To further evaluate if the prior ambiguity is mitigated, we compute the F1 score of all the categories and rearrange the F1 scores by the pixel numbers in the training dataset. Finally, we split and average the rearranged F1 scores into 15 groups, *i.e.*, each bar represents the average F1 scores of 10 categories. The results are shown in Fig. 5. For the categories with few pixels in the training dataset, our method can obtain better results than both the no normalization methods and the learnable methods. For instance, in the first group, our method achieves the average F1



(a) Baseline model.

(b) Ours.

Fig. 6. The similarity map among the weights of the classifier trained on the ADE20K dataset of (a) the baseline model and (b) the proposed methods.

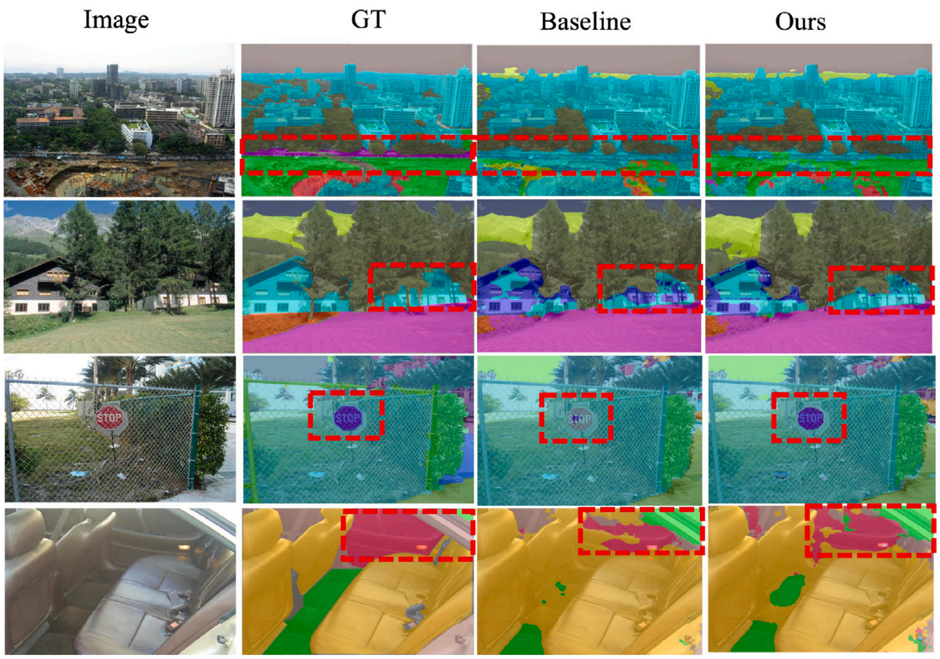


Fig. 7. Visualization of Segformer-B0 baseline and our methods on ADE20K dataset.

Table 9
Experiments on different lengths.

Model	Length	mIoU	mAcc
Segformer-B0	1	34.6	47.6
	2	35.1	48.1
	3	35.3	48.3
	4	35.3	48.2
	5	35.1	48.1

score of 32.0%, which is 7.7% and 2.5% larger than no equalization and learnable methods. The positive correlation between the ℓ_2 norm and the number of training pixels, *i.e.*, the more pixels the larger ℓ_2, makes the network prone to the category with more training pixels and vice versa (“No equalization”, “Learnable”) as shown in Fig. 5. After unifying the ℓ_2 norm (Ours), the network will focus more on the categories with fewer pixels and mitigate the prior ambiguity.

We also conduct experiments to find the relationship between the performance and the length, *i.e.*, ℓ_2 norm, of the proposed CBP as shown in Table 9. We change the length of the CBP from 1 to 5, and the mIoU scores range from 34.6% to 35.3%, and the mAcc ranges from 47.6% to 48.2%. From these experiments, we can find that the length of the CBP is a very sensitive hyperparameter that needs careful design. Besides, we also tested the stabilization of the proposed methods under 4 different random seeds as shown in 10. From the table, we can find that our methods can achieve similar performance in both mIoU and mAcc scores.

Meanwhile, we visualize the similarity maps of our CBP and the learned classifier. We randomly sample some results from the similarity map as shown in Fig. 6(a). The weight of the normal classifier, *i.e.*, optimized by gradient, obtains similar positive similarity among most of the categories, especially for the categories with similar semantics, *e.g.*, oven and microwave. In CBP, however, the similarities between each category turn to **0** as shown in Fig. 6(b), which indicates that the category-basis prototype can solve the problem.

5. Conclusion

In this paper, we propose Category-Basis Prototypes (CBP), a group of **frozen**, **mutually orthogonalized** vectors with **equal** ℓ_2 **norm** to

Table 10
Experiments on different seeds.

Model	Seed	mIoU	mAcc
Segformer-B0	0	35.1	48.1
	1	34.9	47.7
	2	35.2	48.3
	3	34.8	47.7
	mean	35.0 ± 0.2	48.0 ± 0.3

solve the *ambiguity* problem that the prevalent semantic segmentation networks face in the learning of classifier. Moreover, CBP converts the semantic segmentation to an implementation of the EM algorithm where the initialization of CBP can be viewed as the E-step, and the optimization of the segmentation model is seen as the M-step. To better shape the feature shape, we approach Online Centroid Contrastive Loss (OCCL). Specifically, the OCCL consists of two individual losses: centroid loss and category loss. Our experiments show that our methods achieve the expected performance improvements with minimal modification and prove that the CBP, *i.e.*, frozen prototype, performs better than either learnable prototypes or classifiers.

Limitations and Broader Impact. However, there are still lots of limitations in our work. First, several works have shown that multiple prototypes may facilitate the performance of the segmentation model, *e.g.*, ProtoSeg, [5]. Unfortunately, as our CBP is initialized in a novel manner, it is hard to predetermine several groups of prototypes for one category. Second, our methods rely heavily on the number of dimensions in the output features. In specific, if the output dimension is less than the number of categories, our initialization cannot guarantee mutual orthogonality among all prototypes, leading to a violation of our assumptions.

CRediT authorship contribution statement

Jialei Chen: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Daisuke Deguchi:** Writing – review & editing, Writing – original draft, Validation, Supervision, Software, Resources, Project administration, Methodology, Funding acquisition, Conceptualization. **Chenkai Zhang:** Writing – original draft, Methodology, Conceptualization. **Xu Zheng:** Writing – review & editing, Writing – original draft, Validation, Methodology, Conceptualization. **Hiroshi Murase:** Writing – review & editing, Writing – original draft, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by JSPS KAKENHI, Japan Grant Number 23H03474, and JST CREST Grant Number JPMJCR22D1. Computations are done on “Flow” at the Information Technology Center, Nagoya University. The author Jielei Chen is sponsored by the China Scholarship Council.

References

[1] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16 × 16 words: Transformers for image recognition at scale, in: International Conference on Learning Representations, 2021.

[3] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.

[4] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J.M. Alvarez, P. Luo, SegFormer: Simple and efficient design for semantic segmentation with transformers, Adv. Neural Inf. Process. Syst. 34 (2021).

[5] T. Zhou, W. Wang, E. Konukoglu, L. Van Gool, Rethinking semantic segmentation: A prototype view, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 2582–2593.

[6] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick, Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9729–9738.

[7] Z. Wu, Y. Xiong, S.X. Yu, D. Lin, Unsupervised feature learning via non-parametric instance discrimination, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3733–3742.

[8] A.v.d. Oord, Y. Li, O. Vinyals, Representation learning with contrastive predictive coding, 2018, arXiv preprint arXiv:1807.03748.

[9] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ade20k dataset, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 633–641.

[10] H. Caesar, J. Uijlings, V. Ferrari, Coco-stuff: Thing and stuff classes in context, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1209–1218.

[11] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 801–818.

[12] Y. Yuan, X. Chen, J. Wang, Object-contextual representations for semantic segmentation, in: European Conference on Computer Vision, Springer, 2020, pp. 173–190.

[13] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, P. Dollar, R. Girshick, Segment anything, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, 2023, pp. 4015–4026.

[14] F. Li, H. Zhang, P. Sun, X. Zou, S. Liu, J. Yang, C. Li, L. Zhang, J. Gao, Semantic-sam: Segment and recognize anything at any granularity, 2023, arXiv preprint arXiv:2307.04767.

[15] Q. Shen, X. Yang, X. Wang, Anything-3d: Towards single-view anything reconstruction in the wild, 2023, arXiv preprint arXiv:2304.10261.

[16] Y. Yang, X. Wu, T. He, H. Zhao, X. Liu, SAM3D: Segment anything in 3D scenes, 2023, arXiv preprint arXiv:2306.03908.

[17] C. Liang, W. Wang, J. Miao, Y. Yang, Gmmseg: Gaussian mixture based generative semantic segmentation models, Adv. Neural Inf. Process. Syst. 35 (2022) 31360–31375.

[18] S. Liu, J. Ye, R. Yu, X. Wang, Slimmable dataset condensation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 3759–3768.

[19] S. Liu, K. Wang, X. Yang, J. Ye, X. Wang, Dataset distillation via factorization, Adv. Neural Inf. Process. Syst. 35 (2022) 1100–1113.

[20] X. Yang, D. Zhou, S. Liu, J. Ye, X. Wang, Deep model reassembly, Adv. Neural Inf. Process. Syst. 35 (2022) 25739–25753.

[21] Y. Jing, C. Yuan, L. Ju, Y. Yang, X. Wang, D. Tao, Deep graph reprogramming, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 24345–24354.

[22] Y. Jing, Y. Yang, X. Wang, M. Song, D. Tao, Amalgamating knowledge from heterogeneous graph neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 15709–15718.

[23] Z. Lu, S. He, X. Zhu, L. Zhang, Y.-Z. Song, T. Xiang, Simpler is better: Few-shot semantic segmentation with classifier weight transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 8741–8750.

[24] Z. Lu, S. He, D. Li, Y.-Z. Song, T. Xiang, Prediction calibration for generalized few-shot semantic segmentation, IEEE Trans. Image Process. (2023).

[25] Z. Lu, D. Li, Y.-Z. Song, T. Xiang, T.M. Hospedales, Uncertainty-aware source-free domain adaptive semantic segmentation, IEEE Trans. Image Process. (2023).

[26] X. Chen, K. He, Exploring simple siamese representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 15750–15758.

[27] P. Bojanowski, A. Joulin, Unsupervised learning by predicting noise, in: International Conference on Machine Learning, PMLR, 2017, pp. 517–526.

[28] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1026–1034.

[29] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PMLR, 2021, pp. 8748–8763.

[30] H. Hu, J. Cui, L. Wang, Region-aware contrastive learning for semantic segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 16291–16301.

[31] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, A. Joulin, Unsupervised learning of visual features by contrasting cluster assignments, Adv. Neural Inf. Process. Syst. 33 (2020) 9912–9924.

[32] H. Ding, X. Jiang, B. Shuai, A.Q. Liu, G. Wang, Semantic correlation promoted shape-variant context for segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 8885–8894.

[33] B. Cheng, A. Schwing, A. Kirillov, Per-pixel classification is not all you need for semantic segmentation, Adv. Neural Inf. Process. Syst. 34 (2021) 17864–17875.

[34] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3146–3154.

[35] Y. Yuan, R. Fu, L. Huang, W. Lin, C. Zhang, X. Chen, J. Wang, Hrformer: High-resolution vision transformer for dense predict, Adv. Neural Inf. Process. Syst. 34 (2021) 7281–7293.

[36] X. Li, Y. Yang, Q. Zhao, T. Shen, Z. Lin, H. Liu, Spatial pyramid based graph reasoning for semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 8950–8959.

[37] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, J. Sun, Unified perceptual parsing for scene understanding, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 418–434.

[38] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 10012–10022.

[39] W. Wang, C. Han, T. Zhou, D. Liu, Visual recognition with deep nearest centroids, in: The Eleventh International Conference on Learning Representations, 2023.

[40] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, S. Xie, A convnet for the 2020s, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11976–11986.

Jialei Chen received the B.Eng. and M.Eng. degrees from Northeastern University, Shenyang, China in 2019 and 2022. He is currently pursuing the Ph.D. degree in information science from Nagoya University, Japan. His main research interests include semantic segmentation and image processing.

Daisuke Deguchi (Member, IEEE) received the B.Eng. and M.Eng. degrees in engineering and the Ph.D. degree in information science from Nagoya University, Japan, in 2001, 2003, and 2006, respectively. He became a Postdoctoral Fellow at Nagoya University, in 2006. From 2008 to 2012, he was an Assistant Professor at the Graduate School of Information Science. From 2012 to 2019, he was an Associate Professor at Information Strategy Office. Since 2020, he has been an Associate Professor with the Graduate School of Informatics. He is working on the object detection, segmentation, recognition from videos, and their applications to ITS technologies, such as detection and recognition of traffic signs. He is a member of IEICE and IPS Japan.

Chenkai Zhang received the B.Eng. and B.A. degrees from Dalian University of Technology, Dalian, China in 2019, and B.Eng. and M.Eng. degree from Ritsumeikan University, Shiga, Japan in 2019 and 2022. He is currently pursuing a Ph.D. degree in information science from Nagoya University, Japan. His main research interests include explainable artificial intelligence and the reliability of automatic driving.

Xu Zheng (IEEE Student Member) is a Ph.D. student in the Visual Learning and Intelligent Systems Lab, Artificial Intelligence Thrust, The Hong Kong University of Science and Technology, Guangzhou (HKUST-GZ). Before that, he obtained his B.E. and M.S. degree from Northeastern University, Shenyang, China. His research interests lie in computer and robotic vision, multi-modal vision, vision language learning, etc.

Hiroshi Murase (Life Fellow, IEEE) received the B.Eng., M.Eng., and Ph.D. degrees in electrical engineering from Nagoya University, Japan. In 1980, he joined Nippon Telegraph and Telephone Corporation (NTT). From 1992 to 1993, he was a Visiting Research Scientist with Columbia University, New York. He has been a Professor with Nagoya University, since 2003. His research interests include computer vision, pattern recognition, and multimedia information processing. He is a fellow of the IPSJ and the IEICE. He was awarded the IEEE CVPR Best Paper Award, in 1994, the IEEE ICRA Best Video Award, in 1996, the IEICE Achievement Award, in 2002, the IEEE Multimedia Paper Award, in 2004, and the IEICE Distinguished Achievement and Contributions Award, in 2018. He received the Medal with Purple Ribbon from the Government of Japan, in 2012.