Fine-grained Self-supervision for Generalizable Semantic Segmentation

Yuhang Zhang, Shishun Tian, Muxin Liao, Zhengyu Zhang, Wenbin Zou, Chen Xu

Fine-grained Self-supervision for Generalizable Semantic Segmentation

Yuhang Zhang, Shishun Tian, Muxin Liao, Zhengyu Zhang, Wenbin Zou, Chen Xu

***Abstract*—Unsupervised domain adaptative semantic segmentation is a powerful solution for the distribution shift problem between the source and target domains. However, such methods need specified target domain data that may be unavailable in actual applications due to excess expensive collection. Generalizable semantic segmentation as a new paradigm appears in recent research, which aims to generalize well on distinct unseen domains only using source domain data. The existing methods focus on learning domain-invariant features by using global distribution alignment strategies, which may lead to a decreased discriminability of the model. To cope with this challenge, we propose a fine-grained self-supervision (FGSS) framework for generalizable semantic segmentation that takes into account both discriminability and generalizability from the perspective of the intra-class relationship. The FGSS framework contains single-view and multi-view versions. In the single-view version, we propose a fine-grained self-supervision strategy to distinguish the sub-parts of the semantic class for better class discriminability. In the multi-view version, we propose a class prototype feature enhancement strategy to generate another view (i.e. another representation of the original representation). Then, we propose a multi-view mutual supervision loss to enforce consistency between different views and further enhance the generalizability of the model. Experimental results on five widely-used datasets, i.e., GTAV, SYNTHIA, BDD100K, Cityscapes, and Mapillary, demonstrate that our FGSS framework achieves superior performance compared to state-of-the-art methods.**

***Index Terms*—Semantic segmentation, Domain generalization, Fine-grained, Self-supervision, Intra-class relationship**

I. Introduction

Propelled by the swift progress in deep neural networks [1], Semantic segmentation, a crucial task in computer vision, has made remarkable progress in recent years relying on large amounts of annotations and has been broadly used in lots of applications such as autonomous driving [2], [3]. This task infers a semantic category for each pixel of an image. Despite the rapid development of semantic segmentation in a fully-supervised manner, its drawbacks are also evident.

Y. Zhang, S. Tian, M. Liao, are with Guangdong Key Laboratory of Intelligent Information Processing, College of Electronics and Information Engineering, Shenzhen University, Shenzhen, 518060, China (e-mail: zhangyuhang2019@email.szu.edu.cn, stian@szu.edu.cn, liaomuxin2020@email.szu.edu.cn).

Z. Zhang is with the National Institute of Applied Sciences, Rennes, France, and also with the Institute of Electronics and Telecommunications of Rennes Laboratory (e-mail:zhengyu.zhang@insa-rennes.fr).

W. Zou is with Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen Key Laboratory of Advanced Machine Learning and Applications, Institute of Artificial Intelligence and Advanced Communication, College of Electronics and Information Engineering, Shenzhen University, Shenzhen, 518060, China (e-mail: wzou@szu.edu.cn). (Corresponding author: Wenbin Zou.).

C. Xu is with the College of Mathematics and Statistics, Shenzhen University, Shenzhen 518060, China (e-mail: xuchen@szu.edu.cn).

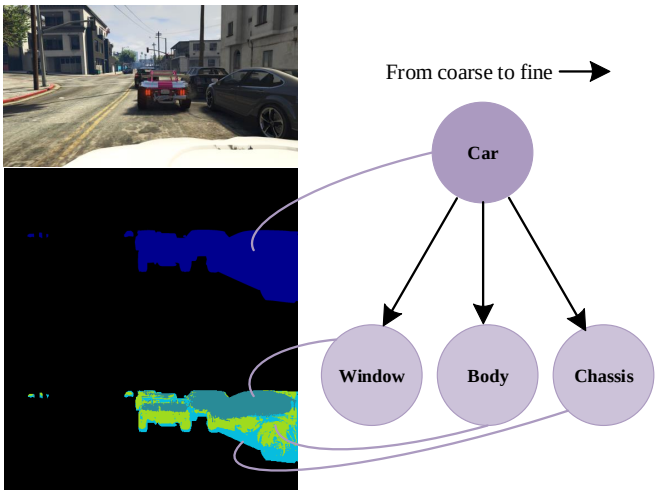

Fig. 1. The illustration of fine-grained segmentation. The pictures from top to bottom are RGB, the car label (coarse-grained label), and the fine-grained label of the car. Fine-grained semantic segmentation segments sub-parts of the coarse semantic category.

First, labeling pixel-level annotation is laborious and time-consuming. As mentioned in GTAV [4], labeling an image from real-world datasets CamVid [5] and Cityscapes [6] takes 60 and 90 minutes, respectively. Consequently, some virtual datasets leverage the computer simulator to collect synthetic data containing automatically generated annotations. The model trained on these synthetic datasets is directly evaluated in the real-world environment. However, there is a discrepancy between the distributions of the training set and the validation set, called the distribution shift problem.

Unsupervised domain adaptation (UDA) technology partially overcomes these drawbacks [7], [8]. Generally, UDA transfers knowledge from a source domain with annotations to a target domain without annotations. It addresses not only the expensive data labeling but also the distribution shift problem in diverse domains, which yields better performance in the target domain. Unfortunately, the target domain data may be unavailable in practical applications. Meanwhile, the UDA model suffers from a dramatic drop in performance when the environment changes.

Domain generalization (DG) is another approach to handle these shortcomings, aiming to transfer knowledge from the source data to all unseen scenes [9], [10] without utilizing the target domains at the training stage. In other words, a DG model is trained using only source domain data and then evaluated on the target domain data. Based on the number

of the source domain, DG can be categorized into single-source DG and multi-source DG, where single-source DG is more challenging than multi-source DG due to its limited domain diversity and less training data. This paper focuses on devising a single-source DG model. Data manipulation and representation learning are two common approaches for DG. The former tries to extend the training set and enhance domain diversity by changing image style to cover new possible domains, while the latter focuses on domain-invariant representation learning. In some circumstances, both types of approaches are co-existing. For example, DRPC [11] randomized the style of the images using auxiliary datasets and then performed pyramid consistency to learn domain-invariant features. Although some progress have been made in the DG task, the existing approaches [12], [13] encourage global distribution alignment, which may lead to the removal of the discriminative information that helps recognize objects better [14]. Recently, SAN-SAW [15] proposed a semantic-aware alignment method to align the distribution for each class independently, which alleviated the loss of local discriminative information. More recently, [16] pointed out that intra-class has various distributions since there are different meaningful sub-parts in a specific category. Nevertheless, such intra-class relationships are ignored in the existing DG methods. Therefore, it is significant to explore the intra-class relationship for better feature discrimination.

An intuitive idea is to distinguish sub-parts of the coarse semantic class, which belongs to the field of fine-grained recognization. Similarly, fine-grained semantic segmentation aims to assign a fine-grained label for each pixel of an image, where fine-grained labels are sub-parts of the coarse semantic categories. A concrete example is shown in Fig. 1, where a car is grouped into windows, bodies, and chassis. Recently, FGN [17] observed that fine-grained learning is beneficial to enhance the discrimination of coarse-grained classification, which is coincidentally consistent with the intuitive idea. To exploit this assumption, we propose a fine-grained self-supervision (FGSS) framework for generalizable semantic segmentation considering both discriminability and generalizability. The FGSS framework contains single-view and multi-view versions. In the single-view version, we propose a fine-grained self-supervision strategy (FSS) to enhance the discriminability of the model. The feature of each coarse semantic class is clustered into distinct object sub-parts by online pseudo-label assignment and then supervised by the fine-grained self-supervision loss. To improve the generalizability, we extend the single-view version to the multi-view version with the proposed Class Prototype Feature Enhancement (CPFE) and Multi-View Mutual Supervision (MVMS). First, CPFE generates an augmented view that can be regarded as a representation with another style but with the same content as the original representation. The augmented features are the intermediate features between the original features and related class prototypes, which are implicitly pulled close to corresponding class prototypes to learn a domain-invariant classifier. Then, MVMS ensures multi-view consistency to learn a generalized representation. In summary, fine-grained segmentation serves as an auxiliary task to improve the generalizability and discriminability for generalized semantic segmentation. This demonstrates for the first time that incorporating fine-grained information is beneficial for generalized semantic segmentation.

Our contributions can be summarized as follows.

- We propose a fine-grained self-supervision framework for generalizable semantic segmentation, which ensures both discriminability and generalizability from the perspective of the intra-class relationships, with both single-view and multi-view versions.
- In the single-view version, we propose a fine-grained self-supervision strategy containing online pseudo-label assignment and a fine-grained self-supervision loss to improve the model discriminability.
- In the multi-view version, we propose the class prototype feature enhancement and multi-view mutual supervision to improve the model generalizability.
- Our proposed FGSS framework outperforms state-of-the-art approaches on several challenging datasets, demonstrating its effectiveness for generalizable semantic segmentation.

The rest of this work is organized as follows. The previous related works are discussed in Section II. Section III describes our FGSS framework in detail. Section IV provides lots of experimental evaluations from many aspects and Section V concludes this work.

II. Related work

A. Semantic segmentation

As a dense prediction task, semantic segmentation plays an important role in computer vision and has emerged in many applications, such as autonomous driving and robots [18]. Fully convolutional networks (FCNs) [19] and the encoder-decoder structure [20] are two types of common basic architectures of semantic segmentation. To incorporate larger context information, feature pyramids were explored in PSPNet [21] and DeepLab-series [22]–[25]. Additionally, self-attention extracted the long-range context by aggregating all pixels of an image, such as DANet [26] and CCNet [27]. More recently, vision transformer based structures [28], [29] have been proposed. Such structures split an image into a sequence of tokens and forward it to Transformer layers for feature extraction. These methods achieved impressive performance but got drastic performance drops when the environment changed or with less supervision. Semantic segmentation with less supervision or the distribution shift problem is still challenging in the future.

B. Unsupervised domain adaptation

Unsupervised domain adaptation (UDA) aims to achieve superior performance in the target domain by reducing the domain gap between the source domain and the target domain [30], [31]. In order to solve the time-consuming data labeling, annotations are used only in the source domain and not in the target domain. The semantic knowledge is transferred from the source domain to the target domain. The existing

UDA methods are roughly parted into three categories, namely appearance adaptation, representation adaptation, and self-training. Changing the image style from the source domain to the target domain is the main idea of the appearance adaptation. Cycada [32] proposed a novel cycle-consistent adversarial adaptation. FDA [33] replaced the low-frequency information of the source images with that of the target images due to the rich domain-invariant information in low-frequency. DACS [34] performed a cross-domain mixing strategy to solve the class conflation problem. Representation adaptation encourages domain adaptation in deep features between the source and target domains by some domain alignment strategies. ASANet [35] and DAST [36] proposed affinity adversarial adaptation and reweighting adversarial adaptation, respectively. ECCA [37] explored more concentrated and consistent activation regions with kernel-based channel attention and mutual information alignment. RCCR [38] proposed a regional contrastive regularization framework to ensure local regional consistency. HDL [39] proposed a hybrid domain learning framework with hybrid domain feature generation and triple domain alignment. DTST [40] proposed stuff and instance matching to improve semantic-level alignment. Prototypical adaptation methods are also the spotlights of representation adaptation, such as CAG [41] and BAPA-Net [42], which aim to pull close the samples and their related prototypes. For self-training, BDL [43] adopted the max probability threshold to generate pseudo labels. To denoise the pseudo label, ProDA [44] proposed prototypical pseudo label denoising. Meanwhile, the self-training strategy was utilized with many UDA methods to furtherly get performance improvements [35], [36], [40]. In spite of that, in the UDA setting, the target domain is assumed to be known and available at the training stage. Such assumptions are indefensible since the target domain is often variable and unavailable in practice.

C. Domain generalization

Domain generalization (DG) seeks a robust model to generalize well on all possible unseen domains. The target domain data does not participate in the training process of the DG model. Data manipulation and representation learning are two main types of approaches to improve generalization. For data manipulation, the source training set is extended as much as possible to cover all unseen domains at the image level by data augmentation or data generation. For instance, DLOW [45] generated a continuous sequence of intermediate domains from one domain to another. DRPC [11] tried to generate images with different styles using auxiliary datasets and then enforced pyramid consistency across data with distinct styles. FSDR [46] randomized image styles in the frequency space by spectrum learning. Besides data manipulation-based methods, the models employing representation learning have been widely devoted due to their effectiveness. For representation learning, the DG model is guided to learn domain-invariant knowledge from different domains or multi-views. IBN-Net [12] aggregated Instance Normalization and Batch Normalization to capture and eliminate appearance variance. ISW [13] proposed an instance selective whitening loss to suppress domain-specific features. SAW-SAN [15] encouraged class-wise semantic consistency by semantic-aware normalization and whitening, which enhanced local feature discrimination. However, the intra-class relationships are ignored in the existing methods. Therefore, it drives us to explore the intra-class relationship to better learn discriminative features.

D. Fine-grained classification

Fine-grained image classification [47] targets classified subordinate categories or sub-parts of the semantic categories. The subordinate categories are presented from a biological perspective, e.g., a dog can be a husky or a teddy. The sub-parts of the semantic classes refer to the different properties of a semantic category, e.g., a car contains some object parts such as the wheel and car light. Compared to the typical category classification, fine-grained classification is more difficult due to similar visuals between the subordinate categories or sub-parts of a semantic class. The key to fine-grained classification is to extract discriminative features. Recently, FGSN [48] proposed a fine-grained segmentation network for visual localization in a self-supervised way. HDNN [49] proposed a hierarchical dilated network with better long-term information flow. RefineMask [50] fused fine-grained features to supplement lost details for high-quality prediction. Chang et al. [17] observed that the fine-grained information helps to improve performances of the coarse classification, which inspired us to propose the fine-grained self-supervision framework. Different from the fine-grained classification task, the fine-grained label in our task is unavailable and there are domain gaps between the training set and the validation set.

E. Self-supervision learning

Self-supervised methods learn meaningful features by various pretext tasks without human annotation to avoid time-consuming data annotations [51]. Self-supervised learning was designed initially as the pre-training process for downstream tasks such as image classification and semantic segmentation. In recent research, self-supervised methods are also aggregated in some downstream tasks to improve the performances of downstream tasks. SEAM [52] proposed an equivariant attention mechanism for weakly supervised semantic segmentation to solve the significant inconsistency of different scale images. SGDepth [53] presented a self-supervised semantically-guided depth estimation to constrain the relationship between the camera pose and geometric projection. SAC [54] proposed a self-supervised augmentation consistency for cross-domain semantic segmentation, which generated pseudo labels by a momentum network allowing stable targets for self-supervised training.

III. Fine-grained self-supervision Framework

A. Problem statement

Given a seen source dataset $\{X_s, Y_s\} \in D_s$ and K unseen target domains $\{X_t^k, Y_t^k\} \in D_t^k, k \in K$, these datasets have a domain gap between each other, where X_* and Y_* are the batch images and corresponding labels from the source

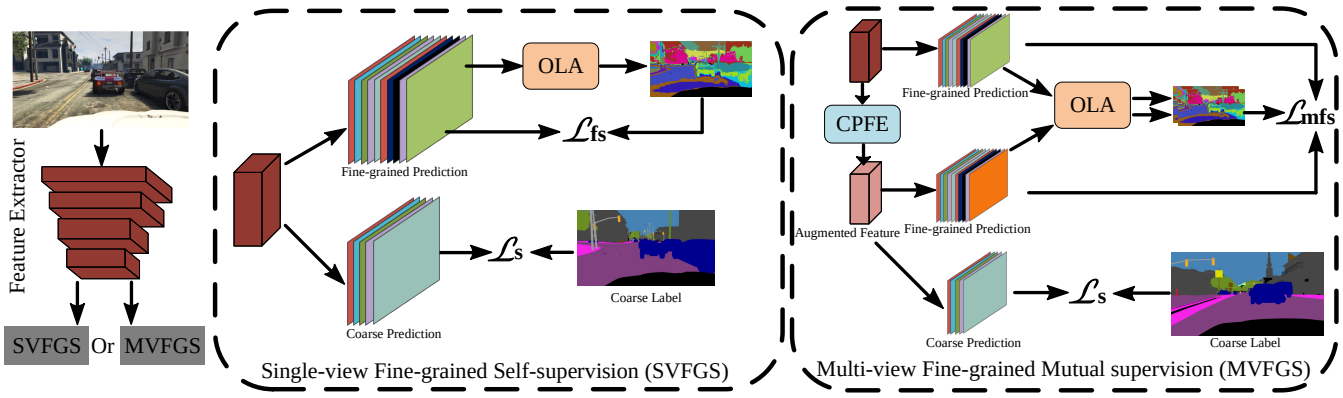

Fig. 2. The pipeline of the proposed FGSS framework containing single-view and multi-view versions. CPFE is class prototype feature enhancement and OLA refers to online pseudo label assignment. Fine-grained segmentation is an auxiliary task to improve the discriminability of the model. Cubes are the deep feature.

and target domains. Different from the unsupervised domain adaptation, the domain generalization task considers the situation where the target domain is entirely unseen and various. In other words, the target domain datasets $D_t^k, k \in K$ are evaluated by the DG model and are not used in the training stage. On the contrary, the source domain data D_s is utilized as the training data. The purpose of the DG task is to perform well on all target domains.

B. Framework overview

As demonstrated in Fig. 2, the fine-grained self-supervision framework has single-view and multi-view versions. Both versions contain an auxiliary task (i.e., fine-grained segmentation) to improve the discriminability of the model. In the multi-view version, the class prototype feature enhancement generates another view (i.e., representation with different styles but the same content as the original representation). The objective of these two versions can be written as:

$$\mathcal{L} = \mathcal{L}_s + \mathcal{L}_{fs} | \mathcal{L}_{mfs} \tag{1}$$

where $\mathcal{L}_s$, $\mathcal{L}_{fs}$, and $\mathcal{L}_{mfs}$ are the coarse segmentation loss, single-view fine-grained self-supervision loss, and multi-view mutual supervision loss, respectively. $|$ is the OR operation.

C. Coarse-grained semantic segmentation

Coarse-grained semantic segmentation has fewer semantic categories than fine-grained semantic segmentation, which is our final goal in this paper. Given a batch of data containing images and labels $\{x_s \in X_s, y_s \in Y_s\}$, the feature f is first extracted by the feature extractor $F(\cdot)$ and then is classified by the coarse classifier C. The cross-entropy is used as coarse segmentation loss $\mathcal{L}_s$, which can be defined as:

$$\mathcal{L}_s = -\sum_{h,w}\sum_{n\in N} y_s^{(h,w,n)} log(C(f^{(h,w)})) \tag{2}$$

where $n \in N$ is the coarse semantic class. $y_s^{(h,w,n)}$ is a one-hot annotation of a pixel at (h, w) position.

D. Fine-grained semantic segmentation

Fine-grained segmentation refers to recognizing the sub-parts or subordinate categories of the semantic categories. The subordinate categories are not applicated in our case. On the contrary, the sub-parts of semantic categories are used reasonably since the object can be divided into different meaningful sub-parts in an image [16]. For example, the person consists of the head, the hand, and the leg visualized in Fig. 8. (c). As pointed out in [17], the fine-grained information helps to improve the discriminability of the coarse segmentation. Consequently, fine-grained semantic segmentation is treated as an auxiliary task to assist the coarse segmentation with a standard cross-entropy L_{fs}, which can be defined as:

$$\mathcal{L}_{fs} = -\sum_{h,w}\sum_{n_f\in N_F} y_{sf}^{(h,w,n_f)} log(C_f(f^{(h,w)})) \tag{3}$$

where $n_f \in N_F$ is the fine-grained class and C_f is the fine-grained classifier. However, fine-grained annotations is not provided in datasets. To tackle this issue, the fine-grained self-supervision strategy (FSS) is proposed to supervise the feature by fine-grained pseudo-label. Thus, the Equation (3) can be rewritten as:

$$\mathcal{L}_{fs} = -\sum_{h,w}\sum_{n_f\in N_F} q_{sf}(y_{sf}|f)^{(h,w,n_f)} log(C_f(f^{(h,w)})) \tag{4}$$

where fine-grained labels y_{sf} are encoded as posterior distributions $q_{sf}(y_{sf}|f)$ by online pseudo-label assignment described in the following paragraph.

E. Online pseudo label assignment

The annotation assignment is an instance of the optimal transport (OT) problem due to the lack of fine-grained annotation. First, the fine-grained predicton $P \in \mathbb{R}^{H\times W\times N_f}$ is extracted by the fine-grained classifier C_f to represent the cost of assigning features $f^{(h,w)}$ to n_f class, which can be defined as:

$$P = C_f(f) = \varphi(Conv(\delta(BN(Conv(f))))) \tag{5}$$

where φ and δ are the Softmax operation and ReLU operation, respectively. $Conv$ and BN are the convolution and batch

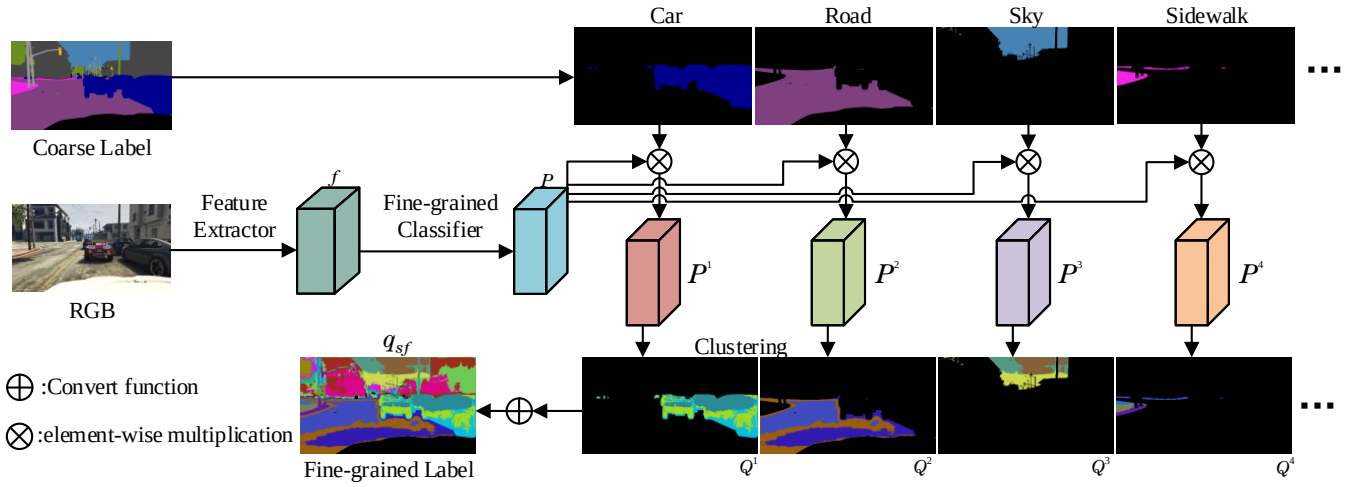

Fig. 3. The structure of online pseudo label assignment, which can be divided into three steps. First, a fine-grained classifier is defined to generate fine-grained prediction P and the fine-grained prediction P^n of the coarse class n is obtained. Second, the Sinkhorn-Knopp algorithm is adopted as the clustering solution on P^n to get the label assignment matrix Q^n of coarse class n. Third, all $\{Q^n, n \in N\} \in Q$ matrixes are converted to the final fine-grained label q_{sf} for self-supervision.

normalization, respectively. To distinguish different sub-parts of each coarse category, the fine-grained prediction belonging to n category P^n is used to perform OT, which is obtained by the element-wise multiplication between the fine-grained prediction and the coarse-grained label. It is denoted as:

$$P^n = P^{(h,w)} \mathbb{1}(y_s^{(h,w,n)} == 1) \tag{6}$$

where $\mathbb{1}$ refers to the indicator function. Then, this OT problem can be fastly solved by the Sinkhorn-Knopp algorithm [55], [56] to guarantee online label assignment at the training stage, where the energy function E_{sk}^n with entropic constraints can be denoted as:

$$\begin{aligned} E_{sk}^n &= \langle Q^n, -logP^n \rangle + \frac{1}{\sigma} KL(Q^n || Rrc^T) \\ s.t. \quad Q^n \mathbf{1}^{K^n} &= \frac{1}{N_c} \mathbf{1}^{N_c}, (Q^n)^\top \mathbf{1}^{N_c} = \frac{1}{K^n} \mathbf{1}^{K^n} \end{aligned} \tag{7}$$

where $Q^n \in \mathbb{R}^{N_c \times K^n}$ is the label assignment matrix. r and c are the marginal projections respectively onto its clusters and feature indices. N_c is the number of clusters for each coarse class, i.e., the number of sub-parts for a coarse class. K^n is the pixel number of n class. $\langle \cdot \rangle$ is the Frobenius dot-product and KL is the Kullback-Leibler divergence. R is a permutation matrix matching clusters to marginals, which represents an arbitrary prior distributions, since the occupied frequencies of the object sub-parts in an image are different [57]. σ is a constant. With the entropic regularization term, the solver in Equation (7) can be rewritten as:

$$Q^n = Diag(u)(P^n)^\sigma Diag(v) \tag{8}$$

where u and v are two renormalization vectors for scaling. After a few iterations, the renormalization vectors are updated to obtain the final assignment matrix Q^n. Finally, the fine-grained label q_{sf} in Equation (4) is converted by these label assignment matrices, which can be denoted as:

$$q_{sf} = \sum_{n=0}^{n \in N} Reshape(Q^n + N_c \times n) \tag{9}$$

where $Reshape$ operation is to match the resolution of the deep feature. Note that the label assignment process does not involve back-propagation. To understand this process clearly, the structure of the online pseudo-label assignment is presented in Fig. 3.

F. Class prototype feature enhancement

The above self-supervised strategy is performed in a single view. Inspired by the DRPC [11] that generated and aligned multi-view data (i.e., representations with different styles but the same content) for better learning domain-invariant features, the multi-view version of the FGSS framework is extended. Generally, another view is often generated by the photometric transform at the image level as SAC [54] does. However, such methods occupy double memory due to feature extraction twice compared to the single view. To reduce memory cost, we propose the class prototype feature enhancement (CPFE) to generate a new view, which focuses on performing perturbation on the original feature rather than on the image level.

1) Class prototypes generation: Class prototypes should be calculated first before conducting feature enhancement. Class prototypes refer to the class-wise feature centroids and are generated online [44]. Given a mini-batch source image $x_s \in X_s$ and a CNN-Based feature extractor $F(\cdot)$, we can obtain the deep feature of an image $F(x_s)$. After that, the prototype of n class p^n can be initialized as:

$$p^n = \frac{\sum_{x_s \in X_s} \sum_h \sum_w F(x_s)^{(h,w)} \mathbb{1}(y_s^{(h,w,n)} == 1)}{\sum_{x_s \in X_s} \sum_h \sum_w \mathbb{1}(y_s^{(h,w,n)} == 1)} \tag{10}$$

To obtain more powerful and generalized class prototypes, the class prototypes are updated using the moving average in two mini-batches, which are defined as,

$$p^n \leftarrow \lambda p^n + (1 - \lambda) p'^n \tag{11}$$

where p'^n is the mean feature of class n calculated by Equation (10) at the current training state and λ is the momentum term that equals 0.9.

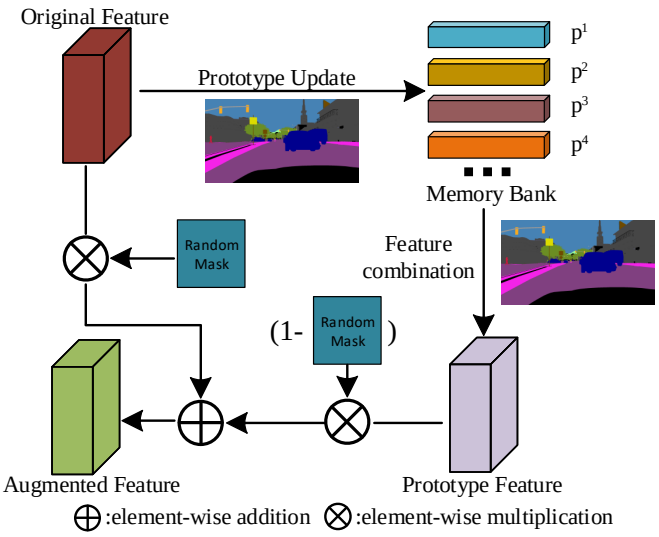

Fig. 4. The structure of the class prototype feature enhancement. The feature combination fills the prototype to a corresponding position according to the ground truth.

2) Feature enhancement: As pointed out in [58], [59], the latent feature contains style and content information. Since the deep feature of the n class f^n and the prototype of n class p^n share the same category, their content information is the same. The discrepancy between them is the style information. Thus, the interpolation between f^n and p^n stands for a new representation (i.e., a new view) with different styles but the same category compared to f^n, which can be denoted as:

$$f^n_{aug} = \lambda_a F(x_s)^{(h,w)} \mathbb{1}(y_s^{(h,w,n)} == 1) + (1 - \lambda_a) p^n \quad (12)$$

where f^n_{aug} is the augmented feature of n class and λ_a is the random value in the range [0-1]. Finally, the augmented feature is put into the coarse classifier C and supervised by the cross-entropy loss as with Equation (2). The structure of the CPFE strategy is shown in Fig. 4.

The reasons for adopting the interpolation to generate another view are as follow. The deep neural network excels at linearizing the feature and decoupling different potential variables linearly [60]–[62]. Meanwhile, the style and semantic information can be linearly separable in deep features [39]. The augmented feature can be regarded as the intermediate feature between the feature and its corresponding prototype. The original feature is augmented toward the class prototype rather than toward a non-meaningful direction. The effect of the CPFE strategy is twofold. For one thing, similar to DLOW [45], this strategy generates the intermediate feature with various styles that plays a data augmentation role. For another, the augmented feature is another view of the same image, which is expected to provide the same content information as the original feature. The deep features of two views are utilized to perform the multi-view mutual supervision introduced in the following paragraph.

G. Multi-view mutual supervision

Fine-grained self-supervision illustrated in Section III. B can be directly used in multi-view, i.e., the label assignment program is performed in two views and then the features of two views are supervised using the fine-grained label generated by themselves respectively. Such naive multi-view self-supervision loss L^n_{mfs} can be denoted as:

$$L^n_{mfs} = L_{fs}(q^{aug}_{sf}, f_{aug}) + L_{fs}(q_{sf}, f) \quad (13)$$

where q_{sf} and q^{aug}_{sf} are the pseudo label generated from the original feature f and augmented feature f_{aug}, respectively. However, the label assignment maps of two views may be inconsistent, which may affect performance. Related experiments are provided in Section IV. D. To handle this issue, a multi-view mutual supervision (MVMS) loss is proposed. The original feature and augmented feature are two views to finish this process. The fine-grained label assignment program is conducted twice to get the label assignment maps of two views, respectively. After that, the fine-grained label of one

Algorithm 1 Fine-grained self-supervision framework

Initialize: $\theta_f, \theta_c, \theta_{fc}, max\_iter, N_c, MultiV$
$iter = 0, Init\_Proto = 0$
Input: Samples $\{ (X_s, Y_s)\}$

for $iter \leq max\_iter$ **do**
  Extracts feature to get f.
  if $Init\_Proto$==1 **then**
    Update prototype by Eq 10 and Eq 11.
  else
    Initilize class prototype by Eq 10.
    $Init\_Proto$ =1
  if $MultiV == 1$ **then**
    Perform CPFE by Eq 12 to get f_{aug}.
    Perform coarse supervision using f_{aug} by Eq 2.
    Perform fine-grained prediction using f and f_{aug} by Eq 5.
    $n = 0$.
    for $n \leq N - 1$ **do**
      Obtain fine-grained predictions P^n and P^n_{aug} by Eq 6.
      Obtain label assignment maps Q^n and Q^n_{aug} by Eq 8.
      $n++$.
    Obtain fine-grained labels q_{sf} and q^{aug}_{sf} by Eq 9.
    Perform MVMS by Eq 14.
  else
    Perform coarse supervision by Eq 2.
    Perform fine-grained prediction by Eq 5.
    $n = 0$.
    for $n \leq N - 1$ **do**
      Obtain P^n by Eq 6.
      Obtain label assignment map Q^n by Eq 8.
      $n++$.
    Obtain the fine-grained label by Eq 9.
    Perform fine-grained supervision by Eq 4.
  Update θ_f, θ_c, θ_{fc}.
  $iter++$.
Return: $\theta_f, \theta_c, \theta_{fc}$.

TABLE I
PERFORMANCE COMPARISON ON RESNET-50 BACKBONE. GTAV (G), BDD (B), SYNTHIA (S), CITYSCAPES (C), AND MAPILLARY (M) ARE ADOPTED AS THE SOURCE DATA IN TURN AND EVALUATED AT OTHER DATASETS. RIGHT ARROW → REPRESENTS GENERALIZING TO ANOTHER DATASET. **BOLD** REPRESENTS THE BEST PERFORMANCE AND UNDERLINE REPRESENTS SECOND-BEST PERFORMANCE. AVG REFERS TO THE AVERAGE MIOU OF ALL EVALUATION SETTINGS.

Methods	Model	Avg	Trained on GTAV (G)				Trained on SYNTHIA (S)				Trained on Cityscapes (C)				Trained on BDD (B)				Trained on Mapillary (M)			
			→C	→B	→M	→S	→C	→B	→M	→G	→B	→M	→G	→S	→G	→S	→C	→M	→G	→S	→C	→B
IBN [12]	ResNet-50	34.2	33.9	32.3	37.8	27.9	32.0	30.6	32.2	26.9	48.6	57.0	45.1	26.1	29.0	25.4	41.1	26.6	30.7	27.0	42.8	31.0
SW [63]	ResNet-50	32.2	29.9	27.5	29.7	27.6	28.2	27.1	26.3	26.5	48.5	55.8	44.9	26.1	27.7	25.4	40.9	25.8	28.5	27.4	40.7	30.5
DRPC [11]	ResNet-50	35.8	37.4	32.1	34.1	28.1	35.7	31.5	32.7	28.8	49.9	56.3	45.6	26.6	33.2	29.8	41.3	31.9	33.0	29.6	46.2	32.9
GTR [64]	ResNet-50	36.1	37.5	33.8	34.5	28.2	36.8	32.0	32.9	28.0	50.8	57.2	45.8	26.5	33.3	30.6	42.6	30.7	32.9	30.3	45.8	32.6
ISW [13]	ResNet-50	36.4	36.6	35.2	40.3	28.3	35.8	31.6	30.8	27.7	50.7	58.6	45.0	26.2	32.7	30.5	43.5	31.6	33.4	30.2	46.4	32.6
SAN-SAW [15]	ResNet-50	38.5	39.8	**37.3**	41.9	30.8	38.9	**35.2**	**34.5**	29.2	**53.0**	**59.8**	**47.3**	28.3	34.8	31.8	44.9	33.2	34.0	31.6	48.7	34.6
PinMem [65]	ResNet-50	41.0	41.2	35.2	39.4	28.9	38.2	32.3	33.9	32.1	50.6	57.9	45.1	29.4	42.4	29.1	54.8	51.0	44.1	30.8	55.9	47.6
Ours	ResNet-50	**42.9**	**44.5**	36.5	**44.3**	**31.4**	**39.2**	29.8	33.5	**34.5**	52.0	58.1	45.1	**30.0**	**47.0**	30.8	**56.2**	**55.5**	**46.7**	**35.5**	**57.4**	**49.8**

view is used to supervise another view. The multi-view mutual supervision loss L_{mfs} can be defined as:

$$L_{mfs} = L_{fs}(q_{sf}, f_{aug}) + L_{fs}(q_{sf}^{aug}, f) \tag{14}$$

The MVMS loss implicitly performs alignment between the original and the augmented views to learn more generalized representation. More details of both versions of the FGSS framework are provided in Algorithm 1.

IV. EXPERIMENT

In this section, we evaluate the effectiveness of our FGSS framework. First of all, the experimental datasets and implemented detail are introduced. Then, the state-of-the-art methods of generalized semantic segmentation are compared to our method. Meanwhile, ablation studies are conducted to verify the influence of proposed components. Finally, segmentation visualization and cluster visualization are provided.

A. Dataset

With the goal of single-domain generalization, the performances of unseen multi-domains need to be evaluated. Five datasets used in the experiments consist of synthetic datasets (GTAV [4] and SYNTHIA [66]) and real-world datasets (Cityscapes [6], BDD100K [67] and Mapillary [68]).

Synthetic datasets: Several synthetic datasets are created to avoid the amount of human effort in labeling. The GTAV dataset collected 25k images from the computer game Grand Theft Auto V with a high resolution of 1914×1052. The SYNTHIA dataset contains 9,400 1280×760 images. 19 and 16 common categories with Cityscapes are used in the GTAV and SYNTHIA datasets, respectively.

Real-world datasets: The Cityscapes dataset is a real-world street scene dataset widely used in the semantic segmentation task, which contains 2975 training and 500 validation images with the resolution of 2048×1024. The BDD100K driving scene dataset contains 7000 training and 1000 validation images with HD resolution (i.e., 1280×720). The Mapillary dataset contains 25k with at least FHD resolution (i.e., 1820×1080), which is captured in various environments such as different weathers and seasons. Both BDD100K and Mapillary datasets adopt 19 overlapped classes with the Cityscapes.

TABLE II
PERFORMANCE COMPARISON ON THE TASK G → {C, B, M} ON SHUFFLENET-V2 AND MOBILENET-V2 BACKBONE NETWORKS. AVG REFERS TO THE AVERAGE MIOU OF ALL EVALUATION SETTINGS.

Model	Method	→C	→B	→M	Avg
ShuffleNet-V2	Baseline	25.6	22.2	28.6	25.4
	IBN-Net [12]	27.1	31.8	34.9	31.3
	ISW [13]	31.0	32.1	35.3	32.8
	DIRL [72]	31.9	32.6	36.1	33.5
	Ours	**35.3**	**33.1**	**37.6**	**35.3**
MobileNet-V2	Baseline	25.9	25.7	26.5	26.0
	IBN-Net [12]	30.1	27.7	27.1	28.3
	ISW [13]	30.9	30.1	30.7	30.5
	DIRL [72]	34.7	32.8	34.3	33.9
	Ours	**35.8**	**33.2**	**36.0**	**35.0**

TABLE III
PERFORMANCE COMPARISON ON THE TASK C → {G, B, S} ON SHUFFLENET-V2 AND MOBILENET-V2 BACKBONE NETWORKS. AVG REFERS TO THE AVERAGE MIOU OF ALL EVALUATION SETTINGS.

Model	Method	→B	→S	→G	Avg
ShuffleNet-V2	Baseline	38.1	21.3	36.5	31.9
	IBN-Net [12]	41.9	23.0	40.9	35.3
	ISW [13]	41.9	22.8	40.2	35.0
	DIRL [72]	42.6	23.7	**41.2**	35.8
	Ours	**44.1**	**27.5**	40.3	**37.3**
MobileNet-V2	Baseline	40.1	21.6	37.3	33.0
	IBN-Net [12]	45.0	23.2	41.1	36.4
	ISW [13]	45.2	22.9	41.2	36.4
	DIRL [72]	**47.6**	23.3	41.4	37.4
	Ours	46.6	**27.6**	**42.3**	**38.8**

B. Implementation Details

The FGSS framework is implemented by the Pytorch library. Three backbones including ResNet-50 [69], ShuffleNet-v2 [70], MobileNet-v2 [71] with DeepLabv3+ architecture [25] are evaluated. The Intersection-over-Union (IoU) of each class and their mean IoU are used as the metric for segmentation accuracy. The model optimization utilizes an SGD optimizer with an initial learning rate of 1e-2, weight decay of 5e-4, and momentum of 0.9. The images are resized to 768×768 resolution and the model is trained in a batch size of 4 with 40k iterations. The cluster number N_c is set to 3 and the related hyperparameter experiment is conducted in Section IV. E. The fine-grained classifier is implemented as a sequence containing 3×3 convolution, batch normalization, ReLU, and 1×1 convolution.

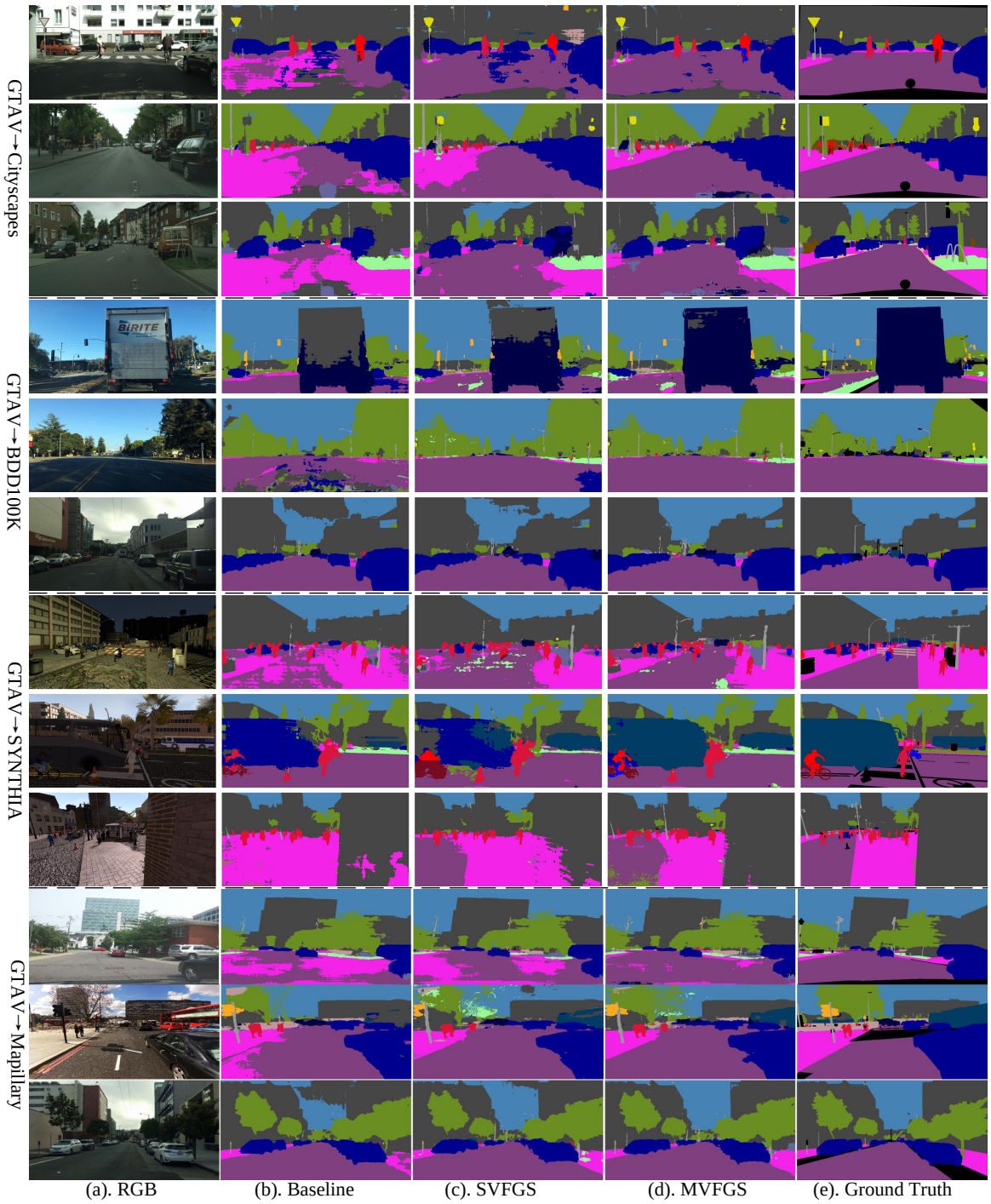

Fig. 5. Qualitative examples. The single-view (SVFGS) and multi-view (MVFGS) versions of the FGSS framework have more right areas in visual.

C. Results

We compare our approach to the other state-of-the-art methods on three backbone networks (i.e., ResNet-50, MobileNet-V2, ShuffleNet-V2). The GTAV, BDD100K, SYNTHIA, Mapillary, and Cityscapes datasets are denoted as G, B, S, M, and C, respectively. In the ResNet-50, these five datasets are taken turns as the source domain for training and the datasets except the source domain are used for evaluation, which is grouped into G $\rightarrow$ {C, B, M, S}, S $\rightarrow$ {G, B, M, C}, C $\rightarrow$ {G, B, S, M}, B $\rightarrow$ {G, C, S, M}, and M $\rightarrow$ {G, B, C, S}. The right arrow $\rightarrow$ refers to "generalizing to". The performance comparison is shown in Table I and the reported results adopt the model of the multi-view version by default. The performance of our method substantially outperforms the second-best methods [65] by 1.9% in terms of average mIoU among the above five generalization settings. In the 20 evaluation settings, our FGSS framework achieves 13 best and 3 second-best performances. Compared with ISW [13], our method achieve 17 better performances in 20 generalization evaluation settings. These results show that our FGSS framework learns a more

TABLE IV
ABLATION STUDIES ON EACH COMPONENT CONTAINING CPFE, FSS, MVMS. THE GTAV DATASET IS USED AS THE SOURCE DOMAIN WITH THE RESNET-50 BACKBONE. AVG REFERS TO THE AVERAGE MIOU OF ALL EVALUATION SETTINGS.

Method	CPFE	FSS	MVMS	→C	→B	→M	→S	Avg
Baseline				36.6	35.2	40.3	28.3	35.1
FSS		✓		41.3	**38.9**	39.8	**32.9**	38.2
CPFE	✓			38.8	36.0	40.0	31.2	36.5
CPFE + FSS	✓	✓		40.3	36.9	39.9	32.3	37.4
CPFE + MVMS	✓		✓	**44.5**	36.5	**44.3**	31.4	**39.2**

TABLE V
COMPLEXITY ANALYSIS ON EACH COMPONENT CONTAINING CPFE, FSS, MVMS. MEMORY REFERS TO THE OCCUPIED MEMORY IN THE TRAINING STAGE, AND TIME IS THE ITERATION TIME FOR ONE IMAGE.

Method	Memory (m)	Time (s)
Baseline	8515	0.30
FSS	8877	0.32
CPFE	8985	0.31
CPFE + FSS	9217	0.39
CPFE + MVMS	9407	0.36

generalized semantic segmentation model. Furthermore, we conduct G → {C, B, M} and C → {G, B, S} on MobileNet-V2 and ShuffleNet-V2 backbone networks. Related results are shown in Table II and Table III. In the task of G → {C, B, M}, both backbone networks achieve the best performance in 3 evaluation settings and outperform the second-best by at least 1.1% in terms of average mIoU. In the task of C → {B, S, G}, compared with DIRL [72], our approach achieves 37.3% and 38.8% average mIoU with a gain 1.5% and 1.4% in the ShuffleNet-V2 and MobileNet-V2 backbone networks, respectively.

Some qualitative examples of different settings (G→C, G→B, G→S, and G→M) are provided in Fig. 5. Some segmentation errors in the Baseline are eliminated in two versions of the FGSS framework. Meanwhile, the multi-view version performs better than the single-view version.

D. Ablation studies

Next, the ablation studies are conducted to investigate the influence of the proposed components including fine-grained self-supervision strategy (FSS), class prototypes feature enhancement (CPFE), and multi-view mutual supervision (MVMS). The single-view version of the FGSS framework (SVFGS) represents FSS. The multi-view version of the FGSS framework (MVFGS) consists of CPFE and MVMS. As shown in Table IV, SVFGS has a clear improvement with a 3.1% in terms of average mIoU compared with the Baseline [13]. Since CPFE generates a new view with various styles and the same semantics as the original view to enlarge available domains, the model improves 1.4% in average mIoU compared with the Baseline. The performance of CPFE + FSS (i.e., naive multi-view self-supervision loss in Equation (13)) is slightly lower than the SVFGS. It is consistent with the opinion illustrated in Section III. E that inconsistent label assignments may affect performance when the FSS strategy is directly used in two

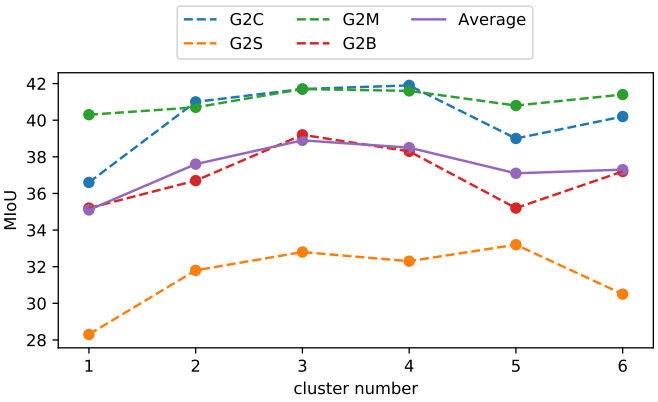

Fig. 6. The effect of the cluster number for each coarse class (i.e., the number of sub-parts for a coarse class).

TABLE VI
PERFORMANCE COMPARISON ON OTHER FEATURE AUGMENTATION METHODS.

Method	→C	→B	→M	→S	Avg
Baseline	36.6	35.2	40.3	28.3	35.1
Mixstyle (random) [73]	37.2	35.5	37.5	30.1	35.1
Mixstyle (cross) [73]	36.8	36.2	39.6	31.0	35.9
Edfmix (random) [74]	38.8	36.3	38.7	30.8	36.2
Edfmix (cross) [74]	40.0	**37.5**	41.0	**32.7**	37.8
Ours	**44.5**	36.5	**44.3**	31.4	**39.2**

views (CPFE + FSS). MVFGS (CPFE + MVMS) achieves 39.2% in average mIoU with a 1.0% gain compared with the SVFGS, which indicates that MVMS learns domain-invariant representation from multi-view.

In addition, the model complexity analysis is also provided. The training time and occupied memory are adopted to analyze the model complexity. As shown in Table V, the multi-view version of the FGSS framework cost 892m memory more and 0.06s more in the training time than the Baseline while getting a 4.1% gain in terms of average mIoU. Note that our framework and the Baseline share the same testing time since the fine-grained branch of our FGSS framework is not used in the evaluation stage. Our method gets significant improvement at low consumption.

E. Cluster number validation

Another concern is the cluster number of each coarse class N_c. Fig. 6 shows the effect of the cluster number. Note that $N_c = 1$ represents the performance of the Baseline. As shown in Fig. 6, the model achieves the best generalizability when N_c equals 3, where the G→B and G→M achieve the best performances compared with other cluster number settings. When N_c is greater than 3, the average mIoU has a decreasing tendency and the performance of the model generalizing on different domains is not stable enough. For example, the performance of G→B is lower than the Baseline when N_c = 5. Moreover, the average mIoU of our framework in different cluster number settings have a performance improvement compared with the Baseline, which shows the effectiveness of our method.

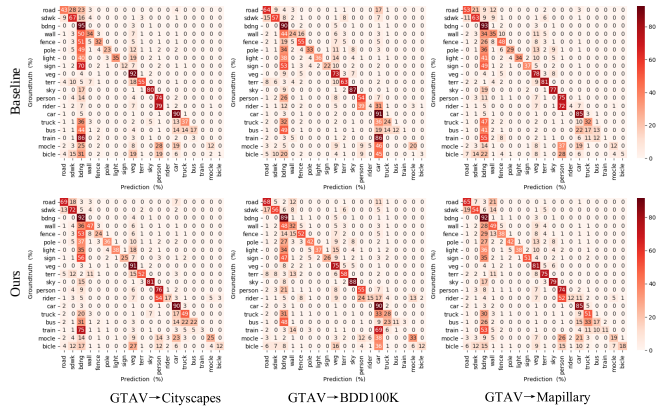

Fig. 7. Confusion matrices comparison on the task of GTAV generalizing to {C, B, M} using ShuffleNet-V2 backbone.

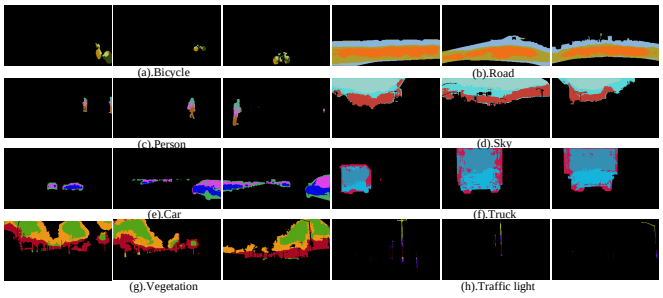

Fig. 8. Some cluster results for different classes. The clustering results with different classes are consistent with human cognition from the visual perspective.

F. Compared with other feature augmentation methods

To verify the advantage of the proposed CPFE, the performance comparison of the feature augmentation methods is shown in Fig. VI. The proposed CPFE achieved the best performance. The augmentation methods designed for the classification task [73], [74] get unsatisfactory performance since semantic segmentation is a pixel-level task simultaneously including several classes in an image instead of including only a class. Directly performing the feature augmentation using the features in the whole spatial location may lead to the class confusion problem. On the contrary, the proposed CPFE generates augmented features at the class level to solve the class confusion problem. The results show the effectiveness of our augmented method.

G. Discrimination validation

As illustrated in Section III, our proposed FGSS framework captures more class discriminative information by distinguishing sub-parts of the semantic categories. An effective manner to verify this viewpoint is to observe the confusion matrix. The confusion matrices generalizing from G to C, B, and M using ShuffleNet-V2 are presented in Fig. 7. The high diagonal value represents the high accuracy of our class predictions. Therefore, the diagonal value is employed as a metric to evaluate the discrimination. Compared with the Baseline, our method achieves 14, 12, and 13 better performances at 19 classes in G→C, B, and M settings. Our proposed FGSS framework performs better in varying categories, showing that our fine-grained self-supervision can enhance the feature discrimination.

TABLE VII
PERFORMANCE ON THE EXTREME CASE (EC).

Method	→C	→B	→M	→S	Avg
Baseline	36.6	35.2	40.3	28.3	35.1
EC	40.6	35.5	41.4	31.1	37.1
SVFGS (Ours)	41.3	**38.9**	39.8	**32.9**	38.2
MVFGS (Ours)	**44.5**	36.5	**44.3**	31.4	**39.2**

H. Cluster visualization

We also present some clustering results with different classes in Fig. 8. The clustering results with different categories are consistent with human cognition from the visual perspective. For example, persons can be divided into head, body, and leg as shown in Fig. 8. (c). Another case is that cars in Fig. 8. (e) consist of windows, bodies, and chassis. The subparts of semantic classes are extracted by label assignment and fine-grained self-supervision is performed to enhance feature discrimination. The visualization of clustering results shows that the FGSS framework can capture the intra-class relationship and ensure discriminability.

I. Extreme case analysis

Given a feature $F \in \mathbb{R}^{H\times W\times C}$ and related annotation $y \in \mathbb{R}^{H\times W\times N}$, the cluster number N_c^n of the coarse class n equals the pixel number N_p^n of coarse class n in the extreme case, where $\sum_n^N N_p^n = H \times W$. At this time, the clustering operation lost its meaning and cannot be conducted because the feature of each spatial position can be treated as a fine-grained label index. Meanwhile, the fine-grained classifier contains large parameters and greatly reduces the efficiency of the algorithm, which is unreasonable and unfeasible. Concretely, the fine-grained classifier is a mapping function from the original feature $F \in \mathbb{R}^{H\times\ W\times C}$ to the fine-grained prediction $P \in \mathbb{R}^{H\times\ W\times(H\times W)}$, which cannot be implemented due to resource limitations. But we still can try to analyze this situation.

For the single-view version of our proposed method, as we cannot capture the intra-class relationship using clustering, the auxiliary loss cannot work and even hurt performance.

For the multi-view version of our proposed method, if we remove the fine-grained classifier (i.e., the feature of a view directly guides the feature of another view), our proposed method degenerates into the feature distribution alignment of different views. As shown in Table VII, the multi-view version of our proposed method in the extreme case still achieves competitive performance with a clear improvement compared to the Baseline.

V. CONCLUSION

In this paper, we designed a fine-grained self-supervision (FGSS) framework for generalizable semantic segmentation,

which considered both the generalizability and discriminability of the model from the intra-class relationship. The FGSS framework consists of single-view and multi-view versions. In the single-view version, we proposed a fine-grained self-supervision strategy which treated the fine-grained segmentation as an auxiliary task to improve feature discrimination. In the multi-view version, we furtherly proposed class prototype feature enhancement to generate another view from the feature space, where the augmented feature is close to the related prototype forcing the classifier to capture more domain-invariant information. Moreover, we proposed multi-view mutual supervision loss in the fine-grained segmentation task for the label inconsistency problem in two views to implicitly reduce the gap between different views. Extensive experimental results showed that our method achieved superior performance compared to other state-of-the-art methods.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under grants 62171294, 62101344, in part by the key Project of DEGP (Department of Education of Guangdong Province) under grants 2018KCXTD027, in part by the Natural Science Foundation of Guangdong Province, China under grants 2022A1515010159, 2020A1515010959, in part by the Key project of Shenzhen Science and Technology Plan under Grant 20220810180617001, in part by the Interdisciplinary Innovation Team of Shenzhen University and in part by the Tencent "Rhinoceros Birds" - Scientific Research Foundation for Young Teachers of Shenzhen University, China.

REFERENCES

[1] C. Yang, H. Luo, G. Liao, Z. Lu, F. Zhou, and G. Qiu, "Self-supervised video super-resolution by spatial constraint and temporal fusion," in *Pattern Recognition and Computer Vision: 4th Chinese Conference, PRCV 2021, Beijing, China, October 29–November 1, 2021, Proceedings, Part III 4*. Springer, 2021, pp. 249–260.

[2] J. Ji, R. Shi, S. Li, P. Chen, and Q. Miao, "Encoder-decoder with cascaded crfs for semantic segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 5, pp. 1926–1938, 2020.

[3] G. Hua, M. Liao, S. Tian, Y. Zhang, and W. Zou, "Multiple relational learning network for joint referring expression comprehension and segmentation," *IEEE Transactions on Multimedia*, 2023.

[4] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proceedings of the European conference on computer vision*, Amsterdam, The Netherlands, 2016, pp. 102–118.

[5] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[6] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Las Vegas, Nevada, USA, 2016, pp. 3213–3223.

[7] Q. Zhou, Z. Feng, Q. Gu, J. Pang, G. Cheng, X. Lu, J. Shi, and L. Ma, "Context-aware mixup for domain adaptive semantic segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[8] W. Zou, R. Long, Y. Zhang, M. Liao, Z. Zhou, and S. Tian, "Dual geometric perception for cross-domain road segmentation," *Displays*, vol. 76, p. 102332, 2023.

[9] Y. Liu, Z. Xiong, Y. Li, X. Tian, and Z.-J. Zha, "Domain generalization via encoding and resampling in a unified latent space," *IEEE Transactions on Multimedia*, 2021.

[10] M. Liao, S. Tian, Y. Zhang, G. Hua, W. Zou, and X. Li, "Domain-invariant information aggregation for domain generalization semantic segmentation," *Neurocomputing*, vol. 546, p. 126273, 2023.

[11] X. Yue, Y. Zhang, S. Zhao, A. Sangiovanni-Vincentelli, K. Keutzer, and B. Gong, "Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 2100–2110.

[12] X. Pan, P. Luo, J. Shi, and X. Tang, "Two at once: Enhancing learning and generalization capacities via ibn-net," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 464–479.

[13] S. Choi, S. Jung, H. Yun, J. T. Kim, S. Kim, and J. Choo, "Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 580–11 590.

[14] X. Jin, C. Lan, W. Zeng, and Z. Chen, "Style normalization and restitution for domain generalization and adaptation," *IEEE Transactions on Multimedia*, 2021.

[15] D. Peng, Y. Lei, M. Hayat, Y. Guo, and W. Li, "Semantic-aware domain generalized segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2594–2605.

[16] T. Zhou, W. Wang, E. Konukoglu, and 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.

[17] D. Chang, K. Pang, Y. Zheng, Z. Ma, Y.-Z. Song, and J. Guo, "Your" flamingo" is my" bird": Fine-grained, or not," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 476–11 485.

[18] G. Li, Y. Wang, Z. Liu, X. Zhang, and D. Zeng, "Rgb-t semantic segmentation with location, activation, and sharpening," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Boston, Ma, USA, 2015, pp. 3431–3440.

[20] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[21] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Honolulu, Hawaii, USA, 2017, pp. 2881–2890.

[22] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.

[23] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[24] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[25] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and 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.

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

[27] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnet: Criss-cross attention for semantic segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Long Beach, CA, USA, 2019, pp. 603–612.

[28] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr *et al.*, "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 6881–6890.

[29] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," *Advances in Neural Information Processing Systems*, vol. 34, pp. 12 077–12 090, 2021.

[30] L. Zhang, P. Wang, W. Wei, H. Lu, C. Shen, A. van den Hengel, and Y. Zhang, "Unsupervised domain adaptation using robust class-wise matching," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 5, pp. 1339–1349, 2018.

[31] Y. Zhao, Z. Zhong, Z. Luo, G. H. Lee, and N. Sebe, "Source-free open compound domain adaptation in semantic segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 10, pp. 7019–7032, 2022.

[32] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell, "Cycada: Cycle-consistent adversarial domain adaptation," in *Proceedings of the International conference on machine learning*, Stockholmsmässan, Stockholm Sweden, 2018, pp. 1989–1998.

[33] Y. Yang and S. Soatto, "Fda: Fourier domain adaptation for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, 2020, pp. 4085–4095.

[34] W. Tranheden, V. Olsson, J. Pinto, and L. Svensson, "Dacs: Domain adaptation via cross-domain mixed sampling," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1379–1389.

[35] W. Zhou, Y. Wang, J. Chu, J. Yang, X. Bai, and Y. Xu, "Affinity space adaptation for semantic segmentation across domains," *IEEE Transactions on Image Processing*, vol. 30, pp. 2549–2561, 2020.

[36] F. Yu, M. Zhang, H. Dong, S. Hu, B. Dong, and L. Zhang, "Dast: Unsupervised domain adaptation in semantic segmentation based on discriminator attention and self-training," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, 2021, pp. 10 754–10 762.

[37] M. Liao, G. Hua, S. Tian, Y. Zhang, W. Zou, and X. Li, "Exploring more concentrated and consistent activation regions for cross-domain semantic segmentation," *Neurocomputing*, 2022.

[38] Q. Zhou, C. Zhuang, R. Yi, X. Lu, and L. Ma, "Domain adaptive semantic segmentation via regional contrastive consistency regularization," in *2022 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2022, pp. 01–06.

[39] Y. Zhang, S. Tian, M. Liao, W. Zou, and C. Xu, "A hybrid domain learning framework for unsupervised semantic segmentation," *Neurocomputing*, 2022.

[40] Z. Wang, M. Yu, Y. Wei, R. Feris, J. Xiong, W.-m. Hwu, T. S. Huang, and H. Shi, "Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, 2020, pp. 12 635–12 644.

[41] Q. ZHANG, J. Zhang, W. Liu, and D. Tao, "Category anchor-guided unsupervised domain adaptation for semantic segmentation," *Advances in Neural Information Processing Systems*, vol. 32, pp. 435–445, 2019.

[42] Y. Liu, J. Deng, X. Gao, W. Li, and L. Duan, "Bapa-net: Boundary adaptation and prototype alignment for cross-domain semantic segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 8801–8811.

[43] Y. Li, L. Yuan, and N. Vasconcelos, "Bidirectional learning for domain adaptation of semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 2019, pp. 6936–6945.

[44] P. Zhang, B. Zhang, T. Zhang, D. Chen, Y. Wang, and F. Wen, "Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 414–12 424.

[45] R. Gong, W. Li, Y. Chen, and L. V. Gool, "Dlow: Domain flow for adaptation and generalization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2477–2486.

[46] J. Huang, D. Guan, A. Xiao, and S. Lu, "Fsdr: Frequency space domain randomization for domain generalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6891–6902.

[47] X.-S. Wei, Y.-Z. Song, O. Mac Aodha, J. Wu, Y. Peng, J. Tang, J. Yang, and S. Belongie, "Fine-grained image analysis with deep learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[48] M. Larsson, E. Stenborg, C. Toft, L. Hammarstrand, T. Sattler, and F. Kahl, "Fine-grained segmentation networks: Self-supervised segmentation for improved long-term visual localization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 31–41.

[49] S. Zhou, D. Nie, E. Adeli, Y. Gao, L. Wang, J. Yin, and D. Shen, "Fine-grained segmentation using hierarchical dilated neural networks," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2018, pp. 488–496.

[50] G. Zhang, X. Lu, J. Tan, J. Li, Z. Zhang, Q. Li, and X. Hu, "Refinemask: Towards high-quality instance segmentation with fine-grained features," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 6861–6869.

[51] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 11, pp. 4037–4058, 2020.

[52] Y. Wang, J. Zhang, M. Kan, S. Shan, and X. Chen, "Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 275–12 284.

[53] M. Klingner, J.-A. Termöhlen, J. Mikolajczyk, and T. Fingscheidt, "Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance," in *European Conference on Computer Vision*. Springer, 2020, pp. 582–600.

[54] N. Araslanov and S. Roth, "Self-supervised augmentation consistency for adapting semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 384–15 394.

[55] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in neural information processing systems*, vol. 26, 2013.

[56] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in neural information processing systems*, vol. 33, pp. 9912–9924, 2020.

[57] Y. Asano, M. Patrick, C. Rupprecht, and A. Vedaldi, "Labelling unlabelled videos from scratch with multi-modal self-supervision," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4660–4671, 2020.

[58] W.-L. Chang, H.-P. Wang, W.-H. Peng, and W.-C. Chiu, "All about structure: Adapting structural information across domains for boosting semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 2019, pp. 1900–1909.

[59] P. Song, L. Dai, P. Yuan, H. Liu, and R. Ding, "Achieving domain generalization in underwater object detection by image stylization and domain mixup," *arXiv preprint arXiv:2104.02230*, 2021.

[60] P. Upchurch, J. Gardner, G. Pleiss, R. Pless, N. Snavely, K. Bala, and K. Weinberger, "Deep feature interpolation for image content changes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Honolulu, Hawaii, USA, 2017, pp. 7064–7073.

[61] Y. Wang, X. Pan, S. Song, H. Zhang, G. Huang, and C. Wu, "Implicit semantic data augmentation for deep networks," *Advances in Neural Information Processing Systems*, vol. 32, pp. 12 635–12 644, 2019.

[62] S. Li, M. Xie, K. Gong, C. H. Liu, Y. Wang, and W. Li, "Transferable semantic augmentation for domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 516–11 525.

[63] X. Pan, X. Zhan, J. Shi, X. Tang, and P. Luo, "Switchable whitening for deep representation learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1863–1871.

[64] D. Peng, Y. Lei, L. Liu, P. Zhang, and J. Liu, "Global and local texture randomization for synthetic-to-real semantic segmentation," *IEEE Transactions on Image Processing*, vol. 30, pp. 6594–6608, 2021.

[65] J. Kim, J. Lee, J. Park, D. Min, and K. Sohn, "Pin the memory: Learning to generalize semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4350–4360.

[66] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Las Vegas, Nevada, USA, 2016, pp. 3234–3243.

[67] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2636–2645.

[68] G. Neuhold, T. Ollmann, S. Rota Bulo, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4990–4999.

[69] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Las Vegas, Nevada, USA, 2016, pp. 770–778.
[70] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 116–131.
[71] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.
[72] Q. Xu, L. Yao, Z. Jiang, G. Jiang, W. Chu, W. Han, W. Zhang, C. Wang, and Y. Tai, "Dirl: Domain-invariant representation learning for generalizable semantic segmentation," 2022.
[73] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with mixstyle," in *Proceedings of the International Conference on Learning Representations*, 2021.
[74] Y. Zhang, M. Li, R. Li, K. Jia, and L. Zhang, "Exact feature distribution matching for arbitrary style transfer and domain generalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 8035–8045.

Yuhang Zhang received the B.Sc. degree from the Guangdong Ocean University, Guangdong, China, in 2019. He is currently pursuing the Ph.D. degree with the College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China.

His research interests focus on computer vision, domain adaptation, semantic segmentation, and deep learning.

Shishun Tian received the B.Sc. degree from Sichuan University, Chengdu, China, in 2012, the M.Sc. degree from Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun, China, in 2015, and the Ph.D. degree from the National Institute of Applied Sciences, Rennes, France, in 2019.

In 2019, he joined Shenzhen University, where he is currently an Assistant Professor. His research interests include image quality assessment, visual perception, bioadjust and machine learning.

Muxin Liao received the B.Sc. and M.Sc. degrees from the Jiangxi Agricultural University, Jiangxi, China, in 2016 and 2019, respectively. He is currently pursuing the Ph.D. degree with the College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China.

His research interests focus on computer vision, domain adaptation, semantic segmentation, and machine learning.

Zhengyu Zhang received the B.E. degree from Guangzhou University, Guangzhou, China, the M.E. degree from Shenzhen University, Shenzhen, China, in 2018 and 2021, respectively. He is currently pursuing the Ph.D. degree with the National Institute of Applied Sciences, Rennes, France, and also with the Institute of Electronics and Telecommunications of Rennes Laboratory. His research interests include image quality assessment and visual perception.

Wenbin Zou received the M.E. degree in software engineering with a specialization in multimedia technology from Peking University, China, in 2010, and the Ph.D. degree from the National Institute of Applied Sciences, Rennes, France, in 2014. From 2014 to 2015, he was a Researcher with the UMR Laboratoire d'informatique Gaspard-Monge, CNRS, and Ecole des Ponts ParisTech, France.

In 2015, he joined Shenzhen University, Shenzhen, China, where he is currently an Associate Professor. His current research interests include saliency detection, object segmentation, and semantic segmentation.

Chen Xu received the B.Sc. and M.Sc. degrees from Xidian University, Xi an, China, in 1986 and 1989, respectively, and the Ph.D. degree from Xi an Jiaotong University, Xi an, in 1992.

In 1992, he joined Shenzhen University, Shenzhen, China, where he is currently a Professor. From 1999 to 2000, he was a Research Fellow with Kansai University, Suita, Japan, and the University of Hawaii, Honolulu, HI, USA, from 2002 to 2003. His current research interests include image processing, intelligent computing, and wavelet analysis.