Calibration-based Dual Prototypical Contrastive Learning Approach for Domain Generalization Semantic Segmentation

Muxin Liao
Guangdong Key Laboratory of Intelligent Information Processing
College of Electronics and Information Engineering
Shenzhen University
Shenzhen, China
liaomuxin2020@email.szu.edu.cn

Shishun Tian
Guangdong Key Laboratory of Intelligent Information Processing
College of Electronics and Information Engineering
Shenzhen University
Shenzhen, China
stian@szu.edu.cn

Yuhang Zhang
Guangdong Key Laboratory of Intelligent Information Processing
College of Electronics and Information Engineering
Shenzhen University
Shenzhen, China
zhangyuhang2019@email.szu.edu.cn

Guoguang Hua
Guangdong Key Laboratory of Intelligent Information Processing
College of Electronics and Information Engineering
Shenzhen University
Shenzhen, China
huaguoguang2021@email.szu.edu.cn

Wenbin Zou*
Guangdong Key Laboratory of Intelligent Information Processing
College of Electronics and Information Engineering
Shenzhen University
Shenzhen, China
wzou@szu.edu.cn

Xia Li
Guangdong Key Laboratory of Intelligent Information Processing
College of Electronics and Information Engineering
Shenzhen University
Shenzhen, China
lixia@szu.edu.cn

ABSTRACT

Prototypical contrastive learning (PCL) has been widely used to learn class-wise domain-invariant features recently. These methods are based on the assumption that the prototypes, which are represented as the central value of the same class in a certain domain, are domain-invariant. Since the prototypes of different domains have discrepancies as well, the class-wise domain-invariant features learned from the source domain by PCL need to be aligned with the prototypes of other domains simultaneously. However, the prototypes of the same class in different domains may be different while the prototypes of different classes may be similar, which may affect the learning of class-wise domain-invariant features. Based on these observations, a calibration-based dual prototypical contrastive learning (CDPCL) approach is proposed to reduce the domain discrepancy between the learned class-wise features and the prototypes of different domains for domain generalization semantic segmentation. It contains an uncertainty-guided PCL (UPCL) and a hard-weighted PCL (HPCL). Since the domain discrepancies of the prototypes of different classes may be different, we propose an uncertainty probability matrix to represent the domain discrepancies of the prototypes of all the classes. The UPCL estimates the uncertainty probability matrix to calibrate the weights of the prototypes during the PCL. Moreover, considering that the prototypes of different classes may be similar in some circumstances, which means these prototypes are hard-aligned, the HPCL is proposed to generate a hard-weighted matrix to calibrate the weights of the hard-aligned prototypes during the PCL. Extensive experiments demonstrate that our approach achieves superior performance over current approaches on domain generalization semantic segmentation tasks. The source code will be released at https://github.com/seabearlmx/CDPCL.

CCS CONCEPTS

• **Computing methodologies** → **Scene understanding**; **Image segmentation**.

KEYWORDS

domain generalization, semantic segmentation, uncertainty-guided prototypical contrastive learning, hard-weighted prototypical contrastive learning

ACM Reference Format:
Muxin Liao, Shishun Tian, Yuhang Zhang, Guoguang Hua, Wenbin Zou, and Xia Li. 2023. Calibration-based Dual Prototypical Contrastive Learning Approach for Domain Generalization Semantic Segmentation. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29–November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3581783.3611792

1 INTRODUCTION

Semantic segmentation plays an important role in multiple real-world applications, such as autonomous driving [10, 24], environment understanding [11, 19, 53], and medical diagnosing [46, 52]. With the rapid development of deep neural networks, supervised semantic segmentation methods [4, 5, 43] have achieved remarkable

*Corresponding author.

Muxin Liao and Shishun Tian contributed equally to this work.

MM '23, October 29–November 3, 2023, Ottawa, ON, Canada.

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3611792

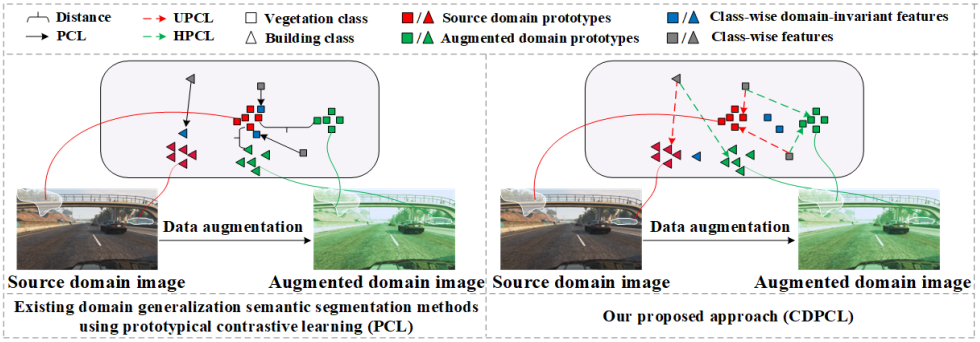


Figure 1: The illustration of the domain discrepancy between the learned class-wise features and the prototypes of different domains. The augmented domain is generated from the source domain by using data augmentation.

progress on the independent and identically distributed assumption. However, the performance of these methods dramatically degrades when they are applied to target domain data, due to the domain discrepancy problem between the training data (source domain) and the testing data (target domain). Collecting abundant target domain images and annotating each pixel for images to retrain the model is one possible solution that is expensive and time-consuming. Thus, unsupervised domain adaptation semantic segmentation (UDASS) methods [20, 54, 58] are proposed to address the domain discrepancy problem.

The key to UDASS methods is to learn domain-invariant features from the labeled source domain and the unlabeled target domain. Although UDASS methods achieve significant performance, they still have limitations. First, UDASS methods could perform well on the target domain but their performances sharply degrade when evaluating out-of-distribution scenes. Second, collecting sufficiently various out-of-distribution data that covers all scenes is impractical and even impossible. To address these limitations, domain generalization methods [39, 41] are proposed.

Domain generalization is a more practical and challenging setting than domain adaptation since any target domain data is not accessed during the training process. Thus, the key to domain generalization is to learn domain-invariant features from single or multiple labeled source domains. Recent methods [2, 13, 22] have been proposed to learn class-wise domain-invariant features from the prototypical contrastive learning (PCL). These methods are based on the assumption that the prototypes, which are represented as the central value of the same class from different domains, are domain-invariant. Since the prototypes of different domains have discrepancies as well [16], the class-wise domain-invariant features learned from the source domain by PCL need to align with the prototypes of the other domains simultaneously. However, the prototypes of the same class in different domains may be different while the prototypes of different classes may be similar [42], which may affect the learning of class-wise domain-invariant features. As shown in the left of Figure 1, the prototypes of the "vegetation" class between the source and augmented domains have discrepancy, where the augmented domain is generated from the source domain by using data augmentation. Thus, there is still domain discrepancy between the class-wise domain-invariant features learned from the source domain by PCL and the prototypes of the augmented domain. Moreover, since the prototypes of the "building" class in the source domain and the prototypes of the "vegetation" class in the augmented domain may be similar, the domain-invariant features of the "vegetation" class learned from the source domain may be close to the "building" class prototypes of the augmented domain. Based on these observations, a calibration-based dual prototypical contrastive learning (CDPCL) approach is proposed to reduce the domain discrepancy between the learned class-wise features and the prototypes of different domains for domain generalization semantic segmentation.

The CDPCL approach contains an uncertainty-guided PCL (UPCL) and a hard-weighted PCL (HPCL). Since the domain discrepancies of the prototypes of different classes may be different [16], we propose an uncertainty probability matrix to represent the domain discrepancies of the prototypes of all the classes. The UPCL estimates the uncertainty probability matrix to calibrate the weights of the prototypes during the PCL. In the uncertainty probability matrix, a small probability means a big difference between the prototypes of different domains. Thus, the uncertainty probability matrix can be set as the weight matrix for the calibration of prototypes. Moreover, considering that the prototypes of different classes may be similar in some circumstances [42], which means these prototypes are hard-aligned [40], the HPCL is proposed to generate a hard-weighted matrix by computing the similarity between the prototypes of different classes for calibrating the weights of the hard-aligned prototypes during the PCL. As shown in the right of Figure 1, the class-wise domain-invariant features learned from our proposed approach are close to the corresponding prototypes of the source and augmented domains and far from the prototypes of different classes. Extensive experiments demonstrate that our approach achieves superior performance over current approaches on domain generalization semantic segmentation. The contributions are summarized as follows:

1. This paper proposes a calibration-based dual prototypical contrastive learning approach to reduce the domain discrepancy between the learned class-wise features and the prototypes of different domains for domain generalization semantic segmentation.

2. We propose an uncertainty-guided prototypical contrastive learning to estimate an uncertainty probability matrix for calibrating the weights of the source domain prototypes during the PCL.
3. We propose a hard-weighted prototypical contrastive learning to generate a hard-weighted matrix for calibrating the weights of the augmented domain prototypes during the PCL.
4. The proposed approach achieves superior performance against the state-of-the-art methods on multiple challenging tasks for domain generalization semantic segmentation.

2 RELATED WORK

2.1 Domain Generalization Methods for Semantic Segmentation

Existing domain generalization semantic segmentation (DGSS) methods are mainly divided into three types: domain randomization methods [36, 56], normalization and whitening methods [6, 28], and meta-learning-based method [15].

Domain randomization methods randomly generate different styles in the input space [29, 37, 50, 56] or the feature space [12, 17, 36] to train a style-insensitive model. For domain randomization methods used in the input space, Yue et al. [50] propose to randomize the style of the source domain images by using CycleGAN [57] to learn style-insensitive features. Different from [50], these methods [29, 37, 56] propose to use a hallucinatory style strategy for randomizing the style of the source domain images. For domain randomization methods used in the feature space, Huang et al. [12] propose to transform the features from the spatial domain to the frequency domain and then randomize the style in the frequency domain. Except for randomizing the style of the source domain images, Lee et al. [17] propose to randomize the content borrowed from the ImageNet for learning class-discriminant features. Different from [12, 17] which perform stylization on coarse-grained image-level features, Su et al. [36] propose a class-aware style variation method to generate fine-grained class-aware stylized images for learning class-level domain-invariant features.

Normalization and whitening methods [6, 26, 27, 45] utilize different style normalization strategies, such as instance normalization [26] or instance selective whitening [6], for learning domain-invariant features. However, normalizing or whitening domain-specific features inevitably remove some task-relevant discriminative information, which may affect the performance. To address this issue, Peng et al. [28] propose semantic-aware normalization and semantic-aware whitening to encourage both intra-category compactness and inter-category separability for enhancing the discriminability of networks. Moreover, based on the meta-learning framework, Kim et al. [15] propose to use an externally settled memory that contains the prototype information of classes for guiding the learning of domain-invariant features.

2.2 Prototypical Contrastive Learning

Prototypical contrastive learning have been utilized in many domain adaptation visual tasks, such as image classification [3], object detection [48, 55], image semantic segmentation [13, 22], and LiDAR point clouds semantic segmentation [49]. These methods claim that the prototypes are beneficial for learning class-wise domain-invariant features since they assume that the prototypes, which are represented as the central value of the same class from different domains, are domain-invariant. Since the prototypes of different domains have discrepancies as well [16], the class-wise domain-invariant features learned from the source domain by PCL need to align with the prototypes of the different domains simultaneously. In the domain adaptation setting, Lee et al. [16] propose a calibration method to compensate for domain discrepancy of prototypes by the distance between the prototypes of the source and target domains. However, the target domain data is not accessed during the training process in the domain generalization setting, which means the prototypes of the target domain are agnostic. Moreover, the prototypes of the same class in different domains may be different while the prototypes of different classes may be similar [42], which may affect the learning of class-wise domain-invariant features.

To address these issues, a calibration-based dual prototypical contrastive learning approach is proposed to reduce the domain discrepancy between the learned class-wise features and the prototypes of different domains for domain generalization semantic segmentation.

2.3 Uncertainty-guided Methods for Domain Generalization

In this section, we discuss recent uncertainty-guided methods for domain generalization. Cai et al. [1] propose a Bayesian CNN-based framework to estimate the model uncertainty for guiding an iterative self-training method. Qiao et al. [31] propose a Bayesian meta-learning framework that aims to increase the capacity of input and label spaces from the source domain by using an uncertainty-guided augmentation strategy. Peng et al. [30] propose an uncertainty-guided domain generalization method to quantify the generalization uncertainty which is used to guide the feature and label augmentation strategies. Xiao et al. [44] propose a probabilistic framework via variational Bayesian inference to learn domain-invariant features by incorporating uncertainty into neural network weights.

Different from these uncertainty-guided domain generalization methods, we propose an uncertainty-guided prototypical contrastive learning to estimate an uncertainty probability matrix for calibrating the weights of the source domain prototypes during the prototypical contrastive learning.

3 APPROACH

3.1 Problem Statement and Overview

Given the labeled dataset as the source domain $S = \{x_i^s, y_i^s\}$, the goal of domain generalization is to train a model on the source domain which performs well on unseen domains. Recent methods [2, 13, 22] aim to learn class-wise domain-invariant features for improving the generalization ability by using prototypical contrastive learning which is denoted as follows:

$$\mathcal{L}_{pcl} = -\sum_{i=1}^{C} y_s^i \log \frac{exp(Z_x^{p_i} \cdot Z_x^{c_i}/\tau)}{\sum_{i \neq k}^{C} exp(Z_x^{p_k} \cdot Z_x^{c_i}/\tau)} \tag{1}$$

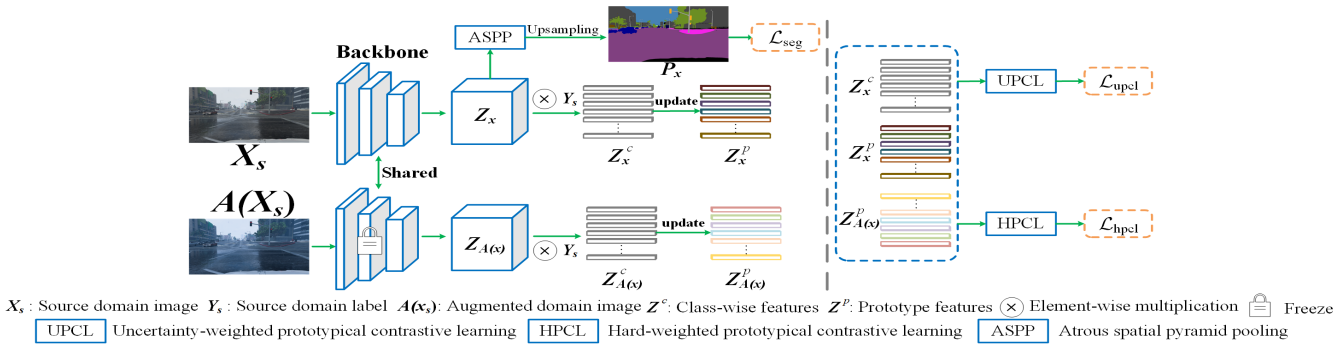


Figure 2: The proposed calibration-based dual prototypical contrastive learning (CDPCL) approach for domain generalization semantic segmentation.

where $Z_x^{p_i}$ is the prototype of the ith class. $Z_x^{c_i}$ is the features of the ith class. τ is the temperature parameter. The class-wise domain-invariant features are learned by minimizing the contrastive learning loss $\mathcal{L}_{pcl}$. The minimized loss means the distance between the $Z_x^{p_i}$ and $Z_x^{c_i}$ which represent the same class is decreased and the distance between the $Z_x^{p_k}$ and $Z_x^{c_i}$ which represent different classes is increased.

Since the prototypes of different domains may have discrepancies as well [16], the class-wise domain-invariant features learned from the source domain by PCL need to align with the prototypes of the different domains simultaneously. However, the prototypes of the same class in different domains may be different while the prototypes of different classes may be similar [42], which may affect the learning of class-wise domain-invariant features. Based on these observations, a calibration-based dual prototypical contrastive learning (CDPCL) approach is proposed to reduce the domain discrepancy between the learned class-wise features and the prototypes of different domains for domain generalization semantic segmentation. The proposed approach is illustrated in Figure 2.

The approach is divided into three parts: a semantic segmentation network, an uncertainty-weighted prototypical contrastive learning (UPCL), and a hard-weighted prototypical contrastive learning (HPCL). Specifically, the semantic segmentation network F is utilized to extract the features Z_x from the source domain images X_s. For one thing, the cross-entropy loss $\mathcal{L}_{seg}$ between the predicted segmentation map P_x and the ground truth Y_s is utilized to optimize the feature Z_x. The cross-entropy loss $\mathcal{L}_{seg}$ is denoted as follows:

$$\mathcal{L}_{seg}(P_x, Y_s) = -Y_s \cdot \log(P_x) \tag{2}$$

where the predicted segmentation map P_x is obtained by upsampling the feature Z_x after putting into the Atrous Spatial Pyramid Pooling module. The Y_s is the ground truth.

For another, the features Z_x are leveraged to generate the prototypes Z_x^p for each class of the source domain. To reduce the domain discrepancy between the class-wise features Z_x^c and the prototypes of the unseen domains, we first construct the augmented prototypes $Z_{A(x)}^p$ from the augmented domains. The augmented domains generated from the source domain by using data augmentation are considered as other unseen domains. Compared with the source domain images, the augmented domain images $A(X_s)$ have the same content but different styles. Then, the augmented images $A(X_s)$ are fed into the backbone to extract the augmented features $Z_{A(x)}$, where the weights of the backbone are frozen. Then, the augmented features $Z_{A(x)}$ are utilized to generate the augmented prototypes $Z_{A(x)}^p$. Finally, the UPCL and HPCL are utilized to reduce the domain discrepancy between the class-wise features Z_x^c and the two prototypes Z_x^p and $Z_{A(x)}^p$. In the following subsections, we sequentially introduce the uncertainty-weighted prototypical contrastive learning, the hard-weighted prototypical contrastive learning, and our total training loss.

3.2 The Uncertainty-weighted Prototypical Contrastive Learning (UPCL)

Since the domain discrepancies of the prototypes of different classes may be different, we propose an uncertainty probability matrix to represent the domain discrepancies of the prototypes of all the classes. The UPCL estimates the uncertainty probability matrix to calibrate the weights of prototypes during the PCL for better learning class-wise domain-invariant features.

Specifically, the difference matrix D is first obtained by using the Manhattan Distance to compute the domain discrepancy between the prototypes Z_x^p and the augmented prototypes $Z_{A(x)}^p$, which is denoted as follows:

$$D(Z_x^p, Z_{A(x)}^p) = ||Z_x^p - Z_{A(x)}^p||_1 \tag{3}$$

where Z_x^p and $Z_{A(x)}^p \in \mathcal{R}^{C\times N}$. The C and N respectively denote the number of classes and the number of features. A big value of D indicates a big difference between the prototypes of the two domains. The prototypes Z_x^p and the augmented prototypes $Z_{A(x)}^p$ are updated in every iteration during the training process, which are denoted as follows:

$$Z_x^p = m_p Z_x^p + (1 - m_p) Z_x^c \tag{4}$$

$$Z_{A(x)}^p = m_a Z_{A(x)}^p + (1 - m_a) Z_{A(x)}^c \tag{5}$$

where m_p and m_a are the trade-off parameters. Z_x^c and $Z_{A(x)}^c$ are the class-wise features respectively obtained from the source domain images and the augmented domain images.

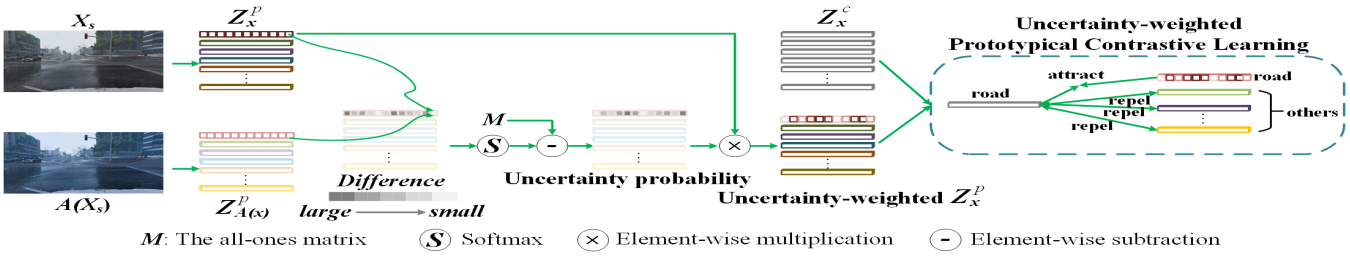


Figure 3: The illustration of the proposed uncertainty-weighted prototypical contrastive learning.

Second, to avoid the class-wise features Z_x^c being aligned with the prototypes with big difference during the PCL, an uncertainty probability matrix is generated to calibrate the weights of these prototypes. The calibration of the prototypes assigns a small weight to the prototypes which have a big difference. To achieve this goals, the uncertainty probability matrix U is obtained by subtracting the probability of the difference matrix D from an all-ones matrix M, which is denoted as follows:

$$U = M - Softmax(D) \tag{6}$$

where $Softmax(D)$ denotes that the probability of the difference matrix D is computed by using the Softmax in the dimensionality of class. In the uncertainty probability matrix U, a small probability means a big difference between the prototypes of the source and augmented domains. Then, the uncertainty probability matrix U is leveraged to calibrate the weights of the prototypes Z_x^p by using element-wise multiplication. Finally, the uncertainty-weighted prototypes are utilized for the UPCL which is denoted as follows by rewriting Eq. (1):

$$\mathcal{L}_{upcl} = -\sum_{i=1}^{C} y_s^i \log \frac{exp((Z_x^{p_i} \times U_i) \cdot Z_x^{c_i}/\tau_u)}{\sum_{i\neq k}^{C} exp((Z_x^{p_k} \times U_k) \cdot Z_x^{c_i}/\tau_u)} \tag{7}$$

where $\times$ means element-wise multiplication. τ_u is the temperature parameter. In particular, the uncertainty probability matrix U is updated when the augmented prototypes $Z_{A(x)}^p$ is updated, which is denoted as follows:

$$U = m_u U + (1 - m_u) U_c \tag{8}$$

where m_u is a trade-off parameter. U_c is the current uncertainty probability matrix.

3.3 The Hard-weighted Prototypical Contrastive Learning (HPCL)

Since the prototypes of different classes may be similar in some circumstances [42], which means these prototypes are hard-aligned [40], the HPCL is proposed to generate a hard-weighted matrix to calibrate the weights of the hard-aligned prototypes during the PCL for better learning class-wise domain-invariant features.

Specifically, first, the similarity matrix of the prototypes between the source and augmented domains is obtained by using cosine similarity, which is denoted as follows:

$$S = \frac{Z_x^p \cdot Z_{A(x)}^p}{||Z_x^p|| \cdot ||Z_{A(x)}^p||} \tag{9}$$

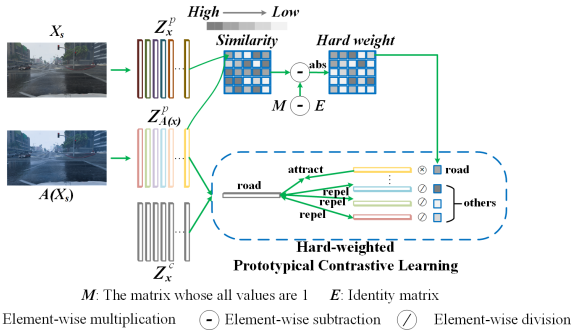


Figure 4: The illustration of the proposed hard-weighted prototypical contrastive learning.

where the similarity matrix $S \in \mathcal{R}^{C\times C}$. In a perfect case, the similarity of the prototypes of the same class between the two domains is the bigger the better and the similarity of the prototypes of different classes between the two domains is the smaller the better. The hard-aligned prototypes are in the opposite case. To calibrate the weights of these prototypes, the hard-weighted matrix H is generated as follows:

$$H = abs(M - E - S) \tag{10}$$

where M is an all-ones matrix and E is an identity matrix. The $abs()$ is to compute the absolute value of a number. In the hard-weighted matrix, a small value means the prototypes of the corresponding class are hard-aligned. Finally, the hard-weighted matrix is utilized to calibrate the prototypes of the augmented domains $Z_{A(x)}^p$ during the PCL. Thus, the HPCL is denoted as follows by rewriting Eq. (1):

$$\mathcal{L}_{hpcl} = -\sum_{i=1}^{C} y_s^i \log \frac{exp(((Z_{A(x)}^{p_i} \times H_{i,i}) \cdot Z_x^{c_i})/\tau_h)}{\sum_{i\neq k}^{C} exp(((Z_{A(x)}^{p_k}/H_{i,k}) \cdot Z_x^{c_i})/\tau_h)} \tag{11}$$

where τ_h is the temperature parameter. When the prototypes are hard-aligned, the $Z_{A(x)}^p$ is weighted with a small value. Thus, the numerator and denominator of the $\mathcal{L}_{hpcl}$ are respectively decreased and increased, which means the loss $\mathcal{L}_{hpcl}$ is increased to optimize the class-wise features for aligning these hard-aligned prototypes.

3.4 The training loss

The overall objective of our approach can be formulated as follows:

$$\mathcal{L} = \mathcal{L}_{seg} + \lambda_1 \mathcal{L}_{upcl} + \lambda_2 \mathcal{L}_{hpcl} \tag{12}$$

Table 1: Performance comparison in terms of mIoU (%) between domain generalization methods in three architectures of the ResNet-50 [9], the ShuffleNetV2 [23], and MobileNetV2 [34] backbones. The best results are marked in bold and the second best results are underlined. † denotes that the performance is obtained by our reproduction of the respective method.

Methods	Backbone	Mean	Train on G				Train on S				Train on C				Train on B				Train on M			
			→C	→B	→M	→S	→C	→B	→M	→G	→B	→M	→G	→S	→G	→S	→C	→M	→G	→S	→C	→B
Deeplabv3+ [5]	ResNet-50	29.9	29.3	25.7	28.3	26.2	23.2	24.5	21.8	26.3	45.2	51.5	42.6	24.3	26.1	21.7	39.0	23.9	25.5	23.4	36.8	26.4
IBN [26]		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 [27]		32.3	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 [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 [29]		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	33.2	30.6	42.6	30.7	32.9	30.3	45.8	32.6
ISW [6]		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 [28]		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 † [15]		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
WildNet † [17]		42.6	**44.6**	38.4	**46.1**	31.3	38.4	33.5	32.8	34.9	50.9	58.8	47.0	28.0	45.1	30.2	55.7	54.1	46.1	30.2	57.1	49.2
SHADE † [56]		42.5	43.5	**40.3**	43.0	31.2	39.6	29.2	34.7	34.8	51.5	58.7	**48.2**	30.8	43.5	31.1	56.2	53.1	45.3	31.1	56.2	48.5
Ours		**45.2**	42.2	39.5	42.4	**33.0**	**41.2**	**35.4**	**35.5**	**36.6**	52.2	**60.7**	46.8	**31.9**	**49.6**	**34.2**	**58.1**	**57.3**	**51.3**	**36.2**	**65.0**	**54.9**
Deeplabv3+ [5]	ShuffleNetV2	33.1	25.6	22.2	28.6	23.3	31.3	22.2	25.9	28.5	38.1	43.3	36.5	25.3	38.8	25.0	47.3	46.8	40.9	25.1	46.6	39.9
IBN [26]		35.5	27.1	31.8	34.9	25.6	32.7	22.8	26.7	30.4	41.9	46.9	40.9	26.5	40.6	25.9	48.6	48.6	42.3	26.6	47.8	42.1
ISW [6]		35.6	31.0	32.1	35.3	24.3	33.7	22.3	26.3	28.8	41.9	47.1	40.2	27.1	40.7	26.3	48.4	48.7	41.5	25.5	48.7	42.4
SAN † [28]		36.3	31.9	30.2	34.8	26.2	32.1	22.3	26.2	28.6	42.3	49.7	38.8	27.6	40.8	28.1	49.8	50.0	42.8	28.1	51.7	43.9
PinMem † [15]		36.1	29.5	31.3	35.4	29.1	32.5	23.0	27.3	30.5	39.9	48.1	37.4	25.9	39.8	**28.5**	49.0	48.3	41.5	30.6	51.5	43.7
SHADE † [56]		37.2	**35.4**	32.3	**36.9**	28.7	**35.4**	23.4	**28.4**	**34.0**	**44.3**	**51.5**	40.9	26.4	**43.0**	26.2	50.6	**51.0**	41.1	26.5	47.2	40.5
Ours		**38.9**	**35.4**	**35.9**	36.3	**30.9**	**35.4**	**24.7**	28.0	32.0	43.6	50.8	**41.1**	**28.1**	41.5	**28.5**	**51.6**	50.5	**46.1**	**31.4**	**56.9**	**48.8**
Deeplabv3+ [5]	MobileNetV2	33.5	25.9	25.7	26.5	24.0	30.1	20.3	22.8	27.5	40.2	44.2	37.8	25.5	40.1	26.6	48.2	48.7	39.1	27.9	47.5	41.6
IBN [26]		35.9	30.1	27.7	27.1	25.0	34.3	20.7	23.6	29.9	45.0	46.9	41.1	27.6	42.7	27.3	52.9	51.3	43.0	29.2	50.7	42.4
ISW [6]		36.4	30.9	30.1	30.7	24.4	34.0	23.5	26.2	29.6	45.2	49.7	41.2	27.2	42.8	28.0	51.5	51.6	39.7	29.3	51.1	41.8
SAN † [28]		37.2	32.5	27.6	30.8	30.4	32.8	21.9	26.6	32.9	45.8	50.1	41.2	26.7	42.5	27.3	51.9	50.8	43.8	31.1	52.5	45.1
PinMem † [15]		37.6	32.2	29.0	31.5	26.5	34.9	23.7	27.9	32.1	46.2	51.9	40.4	26.8	43.5	**31.2**	49.8	51.5	43.6	31.1	52.8	44.5
SHADE † [56]		38.3	34.4	32.4	32.4	27.1	**36.2**	23.8	28.2	33.5	46.1	53.2	41.2	27.2	43.8	30.8	52.6	52.0	42.3	31.0	52.0	45.9
Ours		**40.8**	**36.9**	**33.7**	**36.7**	**31.5**	35.5	**23.9**	**28.4**	**34.3**	**48.4**	**54.2**	**43.2**	**27.3**	**44.6**	30.5	**53.7**	**53.3**	**50.9**	**33.3**	**62.7**	**53.1**

where λ_i is weights coefficients. In particular, during the training process, we freeze the weights of the segmentation network to extract the augmented features $Z_{A(x)}$.

3.5 Behavior in Different Situations

We further discuss the behavior of the proposed approach in different situations. Specifically, we divide the prototype discrepancy in different domains into the following three situations: i) the prototypes of different domains are similar or even the same; ii) the prototypes of different domains may have discrepancies; iii) the prototypes are far away in different domains. In the first situation, all prototypes are assigned with big weights when the prototypes of the source and augmented domains are similar. In particular, the weights approximately equal 1 when the prototypes of different domains are the same. In this situation, the proposed approach can be viewed as the conventional prototypical contrastive learning (PCL). In the second situation, an uncertainty probability matrix and a hard-weighted matrix are generated to calibrate the weights of the prototypes which have a big gap in the source and augmented domains during the prototypical contrastive learning. In the last situation, there is a big gap between the prototypes in the source and augmented domains. All prototypes are assigned with small weights, which means that the class-wise features are pushed away from the prototypes of the source domain to some extent. We argue that it can prevent the model from overfitting the prototypes of the source domain.

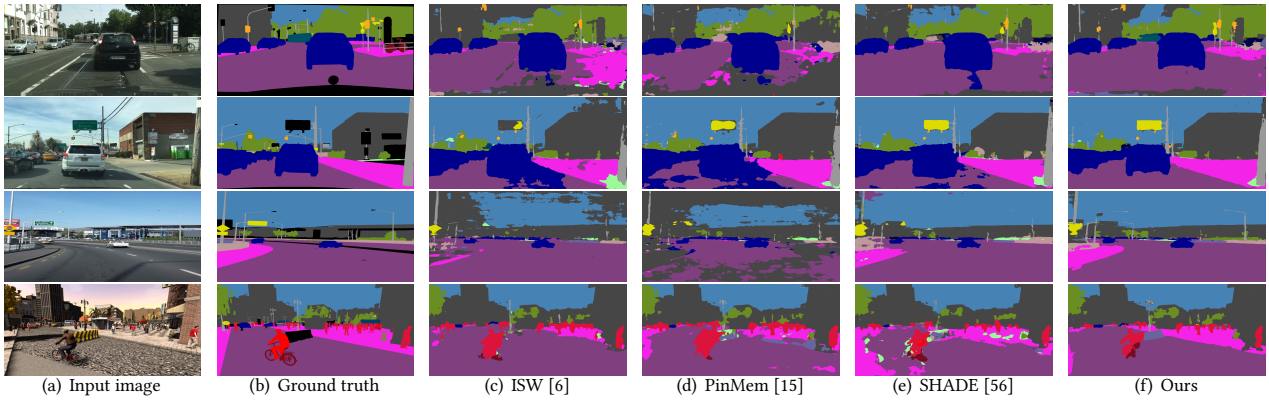
4 EXPERIMENTS

4.1 Datasets and Implementation Details

Five datasets are tested on the proposed approach, including GTA5 (G) [32], SYNTHIA (S) [33], Cityscapes (C) [7], BDD (B) [47], and Mapillary (M) [25]. Our approach is trained on single or several datasets and evaluated on other datasets. In experiments, the ISW [6] is adopted as baseline. The ResNet-50 [9], ShuffleNetV2 [23], and MobileNetV2 [34] are utilized in DeepLabV3+ [5] as the segmentation network. We follow the data augmentations of ISW [6]. Moreover, the weight coefficients m_p, m_a, m_u, τ_u, τ_h, λ_1, and λ_2 are set as 0.9, 0.9, 0.9, 0.8, 0.8, 0.1, and 0.01 for all experiments. More implementation details and the weight coefficient experiments are introduced in the Appendix A.2 and Appendix B.

4.2 Comparison with Single-source Domain Generalization Methods

We compare the performance of our approach with several recent approaches [6, 15, 17, 26–29, 50, 56]. Our approach achieves superior average performance than these methods in five single-source generalization settings on three backbones, which are shown in



(a) Input image (b) Ground truth (c) ISW [6] (d) PinMem [15] (e) SHADE [56] (f) Ours

Figure 5: Visualization comparison between recent methods and our proposed approach in the G→{C, B, M, S} task. The visualization results from four datasets (including Cityscapes, BDD, Mapillary, and SYNTHIA) are respectively shown in four rows.

Table 1. For the ResNet-50 backbone, compared with the SOTA methods SAN [28], PinMem [15], SHADE [56], and WildNet [17], our approach respectively achieves the significant average improvement of 6.7%, 4.2%, 2.7%, and 2.6%. For the ShuffleNetV2 backbone, compared with the SOTA methods SHADE [56] and PinMem [15], our approach achieves an average improvement of 1.7% and 2.8%. For the MobileNetV2 backbone, compared with the SOTA methods SHADE [56] and PinMem [15], our approach achieves an average improvement of 2.5% and 3.2%. In addition, the visualization comparisons of segmentation maps between our proposed approach and recent methods are shown in Figure 5. It shows that the classes of "road", "sidewalk", and "car" are segmented more accurately than recent methods.

Table 2: Performance comparison in terms of mIoU (%) between domain generalization methods in the architecture of the ResNet-50 [9]. The best results are marked in bold and the second-best results are underlined.

Methods	Backbone	Mean	Train on G and S		
			→C	→B	→M
Deeplabv3+ [5]	ResNet-50	30.8	35.5	25.1	31.9
ISW [6]		36.8	37.7	34.1	38.5
MLDG [18]		35.5	38.8	32.0	35.6
PinMem [15]		41.8	44.5	38.1	42.7
SHADE [56]		<u>45.1</u>	<u>47.4</u>	<u>40.3</u>	<u>47.6</u>
Ours		**46.7**	**48.1**	**42.5**	**49.4**

4.3 Comparison with Multi-source Domain Generalization Methods

To further verify the effectiveness of our proposed approach, we compare our proposed approach with recent approaches [6, 15, 18, 56] under the multi-source domain generalization setting. The experimental results are shown in Table 2. As shown in Table 2, our proposed approach respectively achieves the improvement of 4.9% and 1.6% in average mIoU over the SOTA methods PinMem [15] and SHADE [56]. In conclusion, with richer source domains during the training process, our proposed approach can generate more prototypes of different domains and better learn class-wise domain-invariant features from the prototypes of different domains to improve the generalization ability for semantic segmentation. More experiments of multi-source domain generalization settings are shown in the Appendix A.3.

4.4 Ablation Study

In this section, four group experiments are conducted on the "G→{C, B, M, S}" generalization setting to analyze the contributions of each component, including the prototypical contrastive learning (PCL), the uncertainty-weighted prototypical contrastive learning (UPCL), and the hard-weighted prototypical contrastive learning (HPCL), to the final performance and verify our motivation.

The results are given in Table 3. We observe that the performance of the UPCL and HPCL respectively achieve an average improvement of 1.5% and 1.3% than the performance of the PCL. The performance of our proposed approach using the UPCL and HPCL simultaneously is superior to the performance of the PCL, UPCL, and HPCL. It demonstrates that the UPCL and HPCL can both boost the learning of class-wise domain-invariant features. Moreover, we visualize the weight matrix of the learned class-wise features on the source domain and the unseen domain. From Figure 6, we can see that regions are activated by a weight matrix slot corresponding to each class. For the visualization of the PCL, some other classes are activated in the unseen domain (Cityscapes). For example, the class of "sidewalk" is activated in the activation maps of the "road" and "car". the classes of "vegetation" and "sky" are activated in the activation maps of the "building". These results demonstrate that there is still domain discrepancy between the learned class-wise features and the prototypes of the unseen domain. Compared with the visualization of the PCL, the visualization of our proposed approach (CDPCL) achieves a significant improvement. More qualitative and quantitative experiments about the discrepancy change between the learned class-wise features and the different domains before

Table 3: Ablation experiments in the "G→{C, B, M, S}" generalization setting based on the ShuffleNetV2 backbone. The "PCL" indicates the prototypical contrastive learning. The "UPCL" and "HPCL" mean the uncertainty-weighted and the hard-weighted prototypical contrastive learning.

Backbone	Methods	PCL	UPCL	HPCL	Train on GTA5 (G)				
					C	B	M	S	Mean
ShuffleNetV2	Baseline [6]	-	-	-	31.0	32.1	35.3	24.3	30.7
	Ours	√	-	-	33.9	32.5	35.5	28.1	32.5
		-	√	-	34.8	35.3	35.9	29.9	34.0
		-	-	√	35.0	34.1	36.2	29.8	33.8
		-	√	√	**35.4**	**35.9**	**36.3**	**30.9**	**34.6**

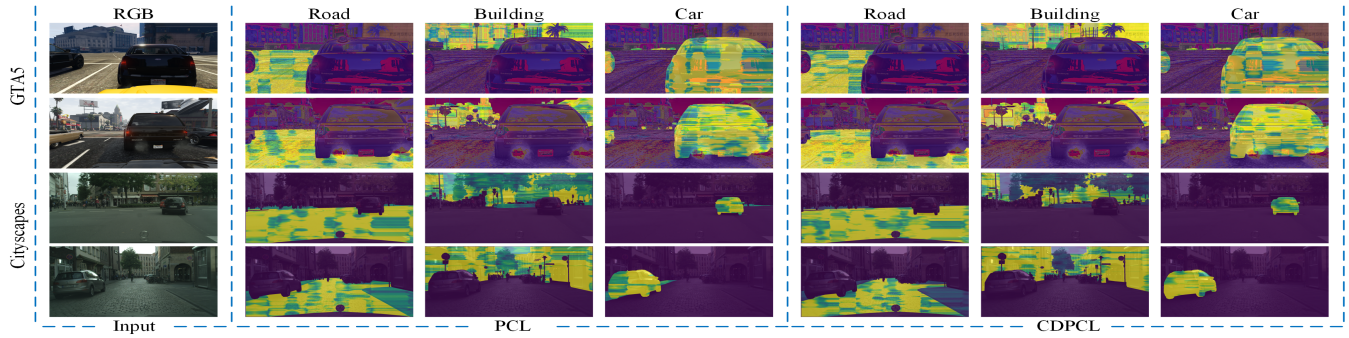


Figure 6: Activation maps visualization comparison of the weight matrix of the learned class-wise features between the PCL and our proposed approach (CDPCL) in the GTA5 to Cityscapes task.

and after applying the proposed approach are analyzed which are shown in the Appendix A.4. In addition, the weights of the activated region corresponding to each class in our proposed approach is higher than the PCL. It demonstrates that our proposed approach can better learn class-wise domain-invariant features. More activation maps visualization and some visualization of the ablation study are shown in the Appendix B.

Table 4: Computational complexity experiments based on the ResNet-50 backbone. Params denotes the number of parameters. FLOPs denotes the number of floating point operations.

Method	FLOPs (G)	Params (M)	Inference time (ms)
baseline [6]	155.92	45.08	131.22
Ours	160.75	47.18	132.80

4.5 Model complexity analysis

We analyze the computational cost of our proposed approach. The input size is set as $1024 \times 512 \times 3$. The experiments are conducted on a single GTX 3090 GPU. As shown in Table 4, for DeepLabV3+ with the ResNet-50 backbone, our proposed approach increases less than 5% on parameters, floating point operations, and inference time. This shows that our proposed approach achieves a significant improvement in performance with very limited extra computational cost.

5 CONCLUSION

In this paper, a calibration-based dual prototypical contrastive learning (CDPCL) approach is proposed to reduce the domain discrepancy between the learned class-wise features and the prototypes of different domains for domain generalization semantic segmentation. The CDPCL approach contains an uncertainty-guided prototypical contrastive learning (UPCL) and a hard-weighted prototypical contrastive learning (HPCL). The proposed UPCL and HPCL respectively generate an uncertainty probability matrix and a hard-weighted matrix to calibrate the weights of the prototypes during the prototypical contrastive learning. Extensive experiments demonstrate that our approach achieves superior performance over current approaches on domain generalization semantic segmentation tasks.

ACKNOWLEDGMENT

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] Minjie Cai, Feng Lu, and Yoichi Sato. 2020. Generalizing hand segmentation in egocentric videos with uncertainty-guided model adaptation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*. 14392–14401.

[2] Chaoqi Chen, Jiongcheng Li, Xiaoguang Han, Xiaoqing Liu, and Yizhou Yu. 2022. Compound domain generalization via meta-knowledge encoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7119–7129.

[3] Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. 2019. Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 627–636.

[4] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence* 40, 4 (2017), 834–848.

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*. 801–818.

[6] Sungha Choi, Sanghun Jung, Huiwon Yun, Joanne T Kim, Seungryong Kim, and Jaegul Choo. 2021. 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*. 11580–11590.

[7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. 2016. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3213–3223.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[10] Yulin He, Wei Chen, Zhengfa Liang, Dan Chen, Yusong Tan, Xin Luo, Chen Li, and Yulan Guo. 2021. Fast and Accurate Lane Detection via Frequency Domain Learning. In *Proceedings of the 29th ACM International Conference on Multimedia*. 890–898.

[11] Guoguang Hua, Muxin Liao, Shishun Tian, Yuhang Zhang, and Wenbin Zou. 2023. Multiple Relational Learning Network for Joint Referring Expression Comprehension and Segmentation. *IEEE Transactions on Multimedia* (2023).

[12] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. 2021. Fsdr: Frequency space domain randomization for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6891–6902.

[13] Zhengkai Jiang, Yuxi Li, Ceyuan Yang, Peng Gao, Yabiao Wang, Ying Tai, and Chengjie Wang. 2022. Prototypical contrast adaptation for domain adaptive semantic segmentation. In *European Conference on Computer Vision*. Springer, 36–54.

[14] Xin Jin, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. 2021. Style normalization and restitution for domain generalization and adaptation. *IEEE Transactions on Multimedia* 24 (2021), 3636–3651.

[15] Jin Kim, Jiyoung Lee, Jungin Park, Dongbo Min, and Kwanghoon Sohn. 2022. Pin the Memory: Learning to Generalize Semantic Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4350–4360.

[16] Geon Lee, Chanho Eom, Wonkyung Lee, Hyekang Park, and Bumsub Ham. 2022. Bi-directional Contrastive Learning for Domain Adaptive Semantic Segmentation. In *European Conference on Computer Vision*. Springer, 38–55.

[17] Suhyeon Lee, Hongje Seong, Seongwon Lee, and Euntai Kim. 2022. WildNet: Learning Domain Generalized Semantic Segmentation from the Wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9936–9946.

[18] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. 2018. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[19] Miaoyu Li, Yachao Zhang, Yuan Xie, Zuodong Gao, Cuihua Li, Zhizhong Zhang, and Yanyun Qu. 2022. Cross-Domain and Cross-Modal Knowledge Distillation in Domain Adaptation for 3D Semantic Segmentation. In *Proceedings of the 30th ACM International Conference on Multimedia*. 3829–3837.

[20] Muxin Liao, Guoguang Hua, Shishun Tian, Yuhang Zhang, Wenbin Zou, and Xia Li. 2022. Exploring More Concentrated and Consistent Activation Regions for Cross-domain Semantic Segmentation. *Neurocomputing* (2022).

[21] Yahao Liu, Jinhong Deng, Xinchen Gao, Wen Li, and Lixin Duan. 2021. Bapa-net: Boundary adaptation and prototype alignment for cross-domain semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8801–8811.

[22] Yulei Lu, Yawei Luo, Li Zhang, Zheyang Li, Yi Yang, and Jun Xiao. 2022. Bidirectional self-training with multiple anisotropic prototypes for domain adaptive semantic segmentation. In *Proceedings of the 30th ACM International Conference on Multimedia*. 1405–1415.

[23] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*. 116–131.

[24] Zeyu Ma, Yang Yang, Guoqing Wang, Xing Xu, Heng Tao Shen, and Mingxing Zhang. 2022. Rethinking Open-World Object Detection in Autonomous Driving Scenarios. In *Proceedings of the 30th ACM International Conference on Multimedia*. 1279–1288.

[25] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. 2017. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE international conference on computer vision*. 4990–4999.

[26] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. 2018. Two at once: Enhancing learning and generalization capacities via ibn-net. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 464–479.

[27] Xingang Pan, Xiaohang Zhan, Jianping Shi, Xiaoou Tang, and Ping Luo. 2019. Switchable whitening for deep representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1863–1871.

[28] Duo Peng, Yinjie Lei, Munawar Hayat, Yulan Guo, and Wen Li. 2022. Semantic-aware domain generalized segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2594–2605.

[29] Duo Peng, Yinjie Lei, Lingqiao Liu, Pingping Zhang, and Jun Liu. 2021. Global and local texture randomization for synthetic-to-real semantic segmentation. *IEEE Transactions on Image Processing* 30 (2021), 6594–6608.

[30] Xi Peng, Fengchun Qiao, and Long Zhao. 2022. Out-of-domain generalization from a single source: An uncertainty quantification approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).

[31] Fengchun Qiao and Xi Peng. 2021. Uncertainty-guided model generalization to unseen domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 6790–6800.

[32] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. 2016. Playing for data: Ground truth from computer games. In *European conference on computer vision*. Springer, 102–118.

[33] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. 2016. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3234–3243.

[34] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4510–4520.

[35] Zu-Yun Shiau, Wei-Wei Lin, Ci-Siang Lin, and Yu-Chiang Frank Wang. 2021. Meta-Learned Feature Critics for Domain Generalized Semantic Segmentation. In *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2244–2248.

[36] Siwei Su, Haijian Wang, and Meng Yang. 2022. Consistency Learning based on Class-Aware Style Variation for Domain Generalizable Semantic Segmentation. In *Proceedings of the 30th ACM International Conference on Multimedia*. 6029–6038.

[37] Gabriel Tjio, Ping Liu, Joey Tianyi Zhou, and Rick Siow Mong Goh. 2022. Adversarial semantic hallucination for domain generalized semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 318–327.

[38] Girish Varma, Anbumani Subramanian, Anoop Namboodiri, Manmohan Chandraker, and CV Jawahar. 2019. IDD: A dataset for exploring problems of autonomous navigation in unconstrained environments. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 1743–1751.

[39] Jingye Wang, Ruoyi Du, Dongliang Chang, Kongming Liang, and Zhanyu Ma. 2022. Domain Generalization via Frequency-domain-based Feature Disentanglement and Interaction. In *Proceedings of the 30th ACM International Conference on Multimedia*. 4821–4829.

[40] Shanshan Wang, Lei Zhang, Pichao Wang, MengZhu Wang, and Xingyi Zhang. 2023. BP-triplet net for unsupervised domain adaptation: A Bayesian perspective. *Pattern Recognition* 133 (2023), 108993.

[41] Yue Wang, Lei Qi, Yinghuan Shi, and Yang Gao. 2022. Feature-based Style Randomization for Domain Generalization. *IEEE Transactions on Circuits and Systems for Video Technology* (2022).

[42] Yinduo Wang, Haofeng Zhang, Zheng Zhang, Yang Long, and Ling Shao. 2020. Learning discriminative domain-invariant prototypes for generalized zero shot learning. *Knowledge-Based Systems* 196 (2020), 105796.

[43] Yanyan Wei, Zhao Zhang, Huan Zheng, Richang Hong, Yi Yang, and Meng Wang. 2022. Sginet: Toward sufficient interaction between single image deraining and semantic segmentation. In *Proceedings of the 30th ACM International Conference on Multimedia*. 6202–6210.

[44] Zehao Xiao, Jiayi Shen, Xiantong Zhen, Ling Shao, and Cees Snoek. 2021. A bit more bayesian: Domain-invariant learning with uncertainty. In *International Conference on Machine Learning*. PMLR, 11351–11361.

[45] Qi Xu, Liang Yao, Zhengkai Jiang, Guannan Jiang, Wenqing Chu, Wenhui Han, Wei Zhang, Chengjie Wang, and Ying Tai. 2022. DIRL: Domain-invariant representation learning for generalizable semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 2884–2892.

[46] Yalan Ye, Ziqi Liu, Yangwuyong Zhang, Jingjing Li, and Hengtao Shen. 2022. Alleviating Style Sensitivity then Adapting: Source-free Domain Adaptation for Medical Image Segmentation. In *Proceedings of the 30th ACM International Conference on Multimedia*. 1935–1944.

[47] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. 2020. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2636–2645.

[48] Jinze Yu, Jiaming Liu, Xiaobao Wei, Haoyi Zhou, Yohei Nakata, Denis Gudovskiy, Tomoyuki Okuno, Jianxin Li, Kurt Keutzer, and Shanghang Zhang. 2022. MTTrans: Cross-domain Object Detection with Mean Teacher Transformer. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IX*. Springer, 629–645.

[49] Zhimin Yuan, Ming Cheng, Wankang Zeng, Yanfei Su, Weiquan Liu, Shangshu Yu, and Cheng Wang. 2023. Prototype-guided Multi-task Adversarial Network for Cross-domain LiDAR Point Clouds Semantic Segmentation. *IEEE Transactions on Geoscience and Remote Sensing* (2023).

[50] Xiangyu Yue, Yang Zhang, Sicheng Zhao, Alberto Sangiovanni-Vincentelli, Kurt Keutzer, and Boqing Gong. 2019. 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*. 2100–2110.

[51] Jian Zhang, Lei Qi, Yinghuan Shi, and Yang Gao. 2022. Generalizable model-agnostic semantic segmentation via target-specific normalization. *Pattern Recognition* 122 (2022), 108292.

[52] Wei Zhang, Xiaohong Zhang, Sheng Huang, Yuting Lu, and Kun Wang. 2022. A Probabilistic Model for Controlling Diversity and Accuracy of Ambiguous Medical Image Segmentation. In *Proceedings of the 30th ACM International Conference on Multimedia*. 4751–4759.

[53] Yachao Zhang, Miaoyu Li, Yuan Xie, Cuihua Li, Cong Wang, Zhizhong Zhang, and Yanyun Qu. 2022. Self-supervised Exclusive Learning for 3D Segmentation with Cross-Modal Unsupervised Domain Adaptation. In *Proceedings of the 30th ACM International Conference on Multimedia*. 3338–3346.

[54] Yuhang Zhang, Shishun Tian, Muxin Liao, Wenbin Zou, and Chen Xu. 2023. A hybrid domain learning framework for unsupervised semantic segmentation. *Neurocomputing* 516 (2023), 133–145.

[55] Yixin Zhang, Zilei Wang, and Yushi Mao. 2021. Rpn prototype alignment for domain adaptive object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 12425–12434.

[56] Yuyang Zhao, Zhun Zhong, Na Zhao, Nicu Sebe, and Gim Hee Lee. 2022. Style-hallucinated dual consistency learning for domain generalized semantic segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*. Springer, 535–552.

[57] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.

[58] Wenbin Zou, Ruijing Long, Yuhang Zhang, Muxin Liao, Zhi Zhou, and Shishun Tian. 2023. Dual geometric perception for cross-domain road segmentation. *Displays* 76 (2023), 102332.

Table 5: Performance comparison in terms of mIoU (%) between domain generalization methods in the architecture of the ResNet-50 [9]. The best results are marked in bold and the second-best results are underlined.

Methods	Backbone	Mean	Train on G, S, and I		
			→C	→B	→M
IBN [26]	ResNet-50	53.1	54.4	48.9	56.1
MLDG [18]		53.1	54.8	48.5	55.9
ISW [6]		53.5	54.7	49.0	56.9
TSMLDG [51]		50.7	53.0	46.4	52.8
PinMem [15]		<u>55.0</u>	<u>56.6</u>	<u>50.2</u>	<u>58.3</u>
Ours		**59.5**	**61.1**	**54.8**	**62.5**

A EXPERIMENTS

A.1 Datasets

Our approach is evaluated on five standard single-source benchmarks and a standard multi-source benchmark. Five standard single-source benchmarks contain "G→{S, C, M, B}", "S→{G, C, M, B}", "C→{G, S, M, B}", "B→{G, S, C, M}", and "M→{G, S, C, B}". The two standard multi-source benchmarks are "{G, S}→{C, B, M}" and "{G, S, I}→{C, B, M}". The "G" and "S" mean GTA5 [32] and SYNTHIA [33] datasets which are two synthetic datasets. The "C", "B", "M", and "I" mean the Cityscapes [7], BDD [47], Mapillary [25], and IDD [38] datasets which are three real-world datasets.

A.2 Implementation Details

In our experiments, the ISW [6] is adopted as baseline. The ResNet-50 [9], ShuffleNetV2 [23], and MobileNetV2 [34] are utilized in DeepLabV3+ [5] as the segmentation network. We follow the data augmentations of previous works, including ISW [6], SAN [28], WildNet [17], PinMemory [15], and SHADE [56]. Specifically, color jittering (brightness of 0.4, contrast of 0.4, saturation of 0.4, and hue of 0.1), Gaussian blur, random cropping, random horizontal flipping, and random scaling with the range of $[0.5, 2.0]$ are used in our approach. The input images of five datasets are cropped to the resolution of 768×768. The mean Intersection-Over-Union value ($mIoU = \frac{TP}{TP+FP+FN}$) is utilized as the metric of evaluation, where TP, FP, and FN are denoted as the predicted pixels numbers of true positive, false positive, and false negative. The Stochastic Gradient Decent (SGD) optimizer with an initial learning rate of $1e-2$ and a momentum of 0.9 is leveraged to optimize our backbone network. In the training stages, the learning rate is adjusted by the power of 0.9 according to the polynomial learning rate scheduler and the maximum number of iterations is set to 40k steps. Moreover, the weight coefficients m_p, m_a, m_u, τ_u, τ_h, λ_1, and λ_2 are set as 0.9, 0.9, 0.9, 0.8, 0.8, 0.1, and 0.01 for all experiments.

A.3 Comparison with Multi-source Domain Generalization Methods

Our proposed approach is trained on three source domains to compare with recent methods, including IBN [26], MLDG [18], ISW [6], TSMLDG [51], and PinMem [15], for further verifying the effectiveness of our proposed approach. As shown in Table 5, our proposed approach respectively achieves an improvement of 4.5% in average mIoU over the PinMem [15]. Thus, with richer source domains during the training process, our proposed approach can better learn class-wise domain-invariant features from the prototypes of different domains to improve the generalization ability for semantic segmentation.

Table 6: The discrepancy between the learned class-wise features and the prototypes of the source domain.

Class	PCL	CDPCL
Road	0.0406	0.0337
Sidewalk	0.0489	0.0504
Building	0.0516	0.0493
Car	0.0331	0.0289

Table 7: The discrepancy between the learned class-wise features and the prototypes of the augmented domain.

Class	PCL	CDPCL
Road	0.0571	0.0458
Sidewalk	0.0494	0.0383
Building	0.0508	0.0462
Car	0.0414	0.0364

A.4 The Analysis of The Distribution Discrepancy Change

We analyze the distribution discrepancy change between the learned class-wise features and the different domains before and after applying the proposed approach from two aspects, including qualitative and quantitative experiments.

Qualitative Results. We visualize activation maps of the weight matrix of the learned class-wise features between the conventional prototypical contrastive learning (PCL) and the proposed approach (CDPCL), which are shown in Figure 6. In the visualization of the PCL, some other classes are activated in the unseen domain (Cityscapes). For example, the classes of "vegetation" and "sky" are activated in the activation maps of the "building". These results demonstrate that there is still domain discrepancy between the learned class-wise features and the prototypes of the unseen domain (Cityscapes). Compared with the visualization of the PCL, the visualization of our proposed approach (CDPCL) achieves a significant improvement. It demonstrates that the difference between the learned class-wise features and the prototypes of the unseen domain (Cityscapes) is reduced.

Quantitative Results. We respectively use the cosine similarity and the Manhattan distance to measure the discrepancy changes between the learned class-wise features and the different domains before and after applying the proposed approach. The discrepancy changes of the "road", "sidewalk", "building", and "car" classes are shown in Table 6, Table 7, Table 8, and Table 9.

First, we compare the discrepancy measured by the Manhattan distance. The discrepancy between the learned class-wise features and the source domains is shown in Table 6. The discrepancy between the learned class-wise features and the augmented domains

Table 8: The similarity of different classes between the learned class-wise features and the source domain prototypes. The first row represents the learned class-wise features and the first column represents the source domain prototypes.

	PCL					CDPCL			
Class	Road	Sidewalk	Building	Car	Class	Road	Sidewalk	Building	Car
Road	0.8598	0.5131	0.4979	0.5368	Road	0.9182	0.4531	0.4157	0.3148
Sidewalk	0.5990	0.8502	0.5972	0.4974	Sidewalk	0.5130	0.8904	0.4691	0.4211
Building	0.5011	0.3995	0.8511	0.3019	Building	0.4853	0.3154	0.9003	0.2117
Car	0.4996	0.5018	0.5810	0.9051	Car	0.3548	0.4183	0.4519	0.9331

Table 9: The similarity of different classes between the learned class-wise features and the augmented domain prototypes. The first row represents the learned class-wise features and the first column represents the augmented domain prototypes.

	PCL					CDPCL			
Class	Road	Sidewalk	Building	Car	Class	Road	Sidewalk	Building	Car
Road	0.7998	0.6131	0.3377	0.4957	Road	0.8512	0.4937	0.2903	0.3304
Sidewalk	0.6982	0.7573	0.3984	0.5965	Sidewalk	0.5701	0.8210	0.2965	0.3899
Building	0.4005	0.3002	0.8007	0.3012	Building	0.2418	0.2419	0.8411	0.1944
Car	0.5190	0.4921	0.2910	0.8793	Car	0.3941	0.3823	0.2740	0.8999

is shown in Table 7. From Table 6, compared with the conventional PCL, the discrepancy between the learned class-wise features and the prototypes of the source domain is reduced in the classes of "road", "building", and "car" by using the proposed approach (CDPCL). From Table 7, compared with the conventional PCL, the discrepancy between the learned class-wise features and the prototypes of the augmented domain is reduced in the classes of "road", "sidewalk", "building", and "car" by using the CDPCL. In addition, although the discrepancy of the "sidewalk" class between the learned class-wise features and the prototypes of the source domain is increased, the discrepancy of the "sidewalk" class between the learned class-wise features and the prototypes of the augmented domain is significantly decreased. We argue that this phenomenon is caused by the big gap between the "sidewalk" class prototypes of the source and augmented domains, which means the "sidewalk" class prototypes of the source domain are uncertain. Thus, a small weight is assigned to the "sidewalk" class prototypes of the source domain.

Second, we compare the discrepancy measured by the cosine similarity. The discrepancy between the learned class-wise features and the source domains is shown in Table 8. The discrepancy between the learned class-wise features and the augmented domains is shown in Table 9. From Table 8 and Table 9, compared with the conventional PCL, the similarity of different classes is reduced while the similarity of the same class is significantly increased by using the proposed approach (CDPCL).

In conclusion, from these qualitative and quantitative experiments, the proposed approach can reduce the discrepancy between the learned class-wise features and the prototypes of different domains.

B SUPPLEMENT MATERIAL

The supplement material will be released at https://github.com/seabearlmx/CDPCL.