Video generalized semantic segmentation via Non-Salient Feature Reasoning and Consistency

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

Video Generalized Semantic Segmentation via Non-salient Feature Reasoning and Consistency

Yuhang Zhang[b,d], Zhengyu Zhang[f], Muxin Liao[b,d], Shishun Tian[b,d], Rong You[e], Wenbin Zou[a,b,c,d,*] and Chen Xu[e]

[a]Shenzhen Key Laboratory of Advanced Machine Learning and Applications, Shenzhen University, Shenzhen, 518060, China
[b]Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen, 518060, China
[c]Institute of Artificial Intelligence and Advanced Communication, Shenzhen University, Shenzhen, 518060, China
[d]College of Electronics and Information Engineering, Shenzhen University, Shenzhen, 518060, China
[e]College of Mathematics and Statistics, Shenzhen University, Shenzhen 518060, China
[f]Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, F-35000 Rennes, France

ARTICLE INFO

Keywords:
Semantic segmentation
Video domain generalization
Non-salient region
Class-wise relationship reasoning
Domain-invariant feature

ABSTRACT

Video semantic segmentation is beneficial for dynamic scene processing in real-world environments, and achieves superior performance on independent and identically distributed data. However, it suffers from performance degradation in environments with various domain styles, which is known as the distribution shift problem. Although some previous studies on image generalized semantic segmentation considered the distribution shift problem, temporal-frame information could not be used to obtain more accurate prediction. Thus, in this study, we explore a new task, known as the video generalized semantic segmentation (VGSS) task, which establishes a connection between continuous frames and domain generalization. We propose a novel method named Non-Salient Feature Reasoning and Consistency (NSFRC) for this task. Specifically, we first define the class-wise non-salient feature, which describes the features of the class-wise non-salient region that carry more generalized information. We then propose a class-wise non-salient feature reasoning strategy to select and enhance generalized channels adaptively. This strategy adopts a new form to use domain-invariant features by treating the domain-invariant features as prior information to assist domain-invariant model learning. Finally, we propose a non-salient centroid alignment loss to alleviate the temporally inconsistent and negative transfer problems in the VGSS task. We also extend our video-based framework to the image generalized semantic segmentation (IGSS) task. Experiments demonstrate that our NSFRC framework yields significant improvements in both the VGSS and IGSS tasks. To explain the idea of this research in a clear and attractive way, we provide the visual abstract shown in Figure 1.

1. Introduction

Semantic segmentation has been employed in many applications such as automatic driving [1], robotics [2] and medicinal diagnosis [3]. It has made significant progress owing to the development of deep learning technology and aims to assign an object class to each pixel of an image [4, 5]. As image-based semantic segmentation cannot use the results of previous frames as prior information to assist in the segmentation of the current frame, some researchers [6, 7] have established the connection between continuous temporal frames for video-based semantic segmentation (VSS). Although these previous studies obtained more accurate segmentation results through inter-frame fusion or consistency, performance degradation occurs in environments with diverse domain styles, which is known as the distribution shift problem.

Unsupervised domain adaptation (UDA) is the preferred technology for handling the distribution shift problem for a single scene, in which the goal is to achieve remarkable performance on the target domain by transferring knowledge from the source domain to the target domain [8, 9]. The source domain with annotations and the target domain without annotations are used simultaneously during the training stage [10]. UDA technology faces two challenges. First, only one scene can be adapted using UDA methods. Second, the target data used for training are not always available for practical applications.

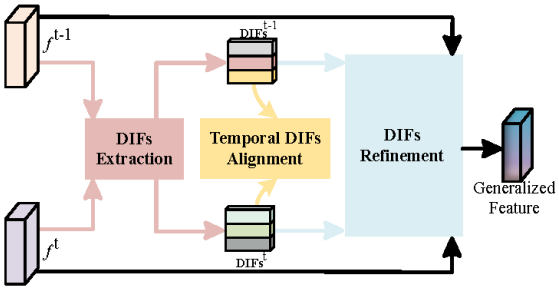

Figure 1: Visual abstract of our method. Domain invariant features (DIFs) extraction, temporal DIFs alignment, and DIFs refinement correspond to the proposed class-wise non-salient feature, non-salient centroid alignment, and non-salient feature reasoning, respectively. The upper right index * ($^{t-1}$ or ^{t}) represents different frames and f is the deep feature.

Domain generalization (DG) is more practical than UDA because it can be adapted to more scenes with diverse domains and the target data are not used during training. The

ORCID(s):

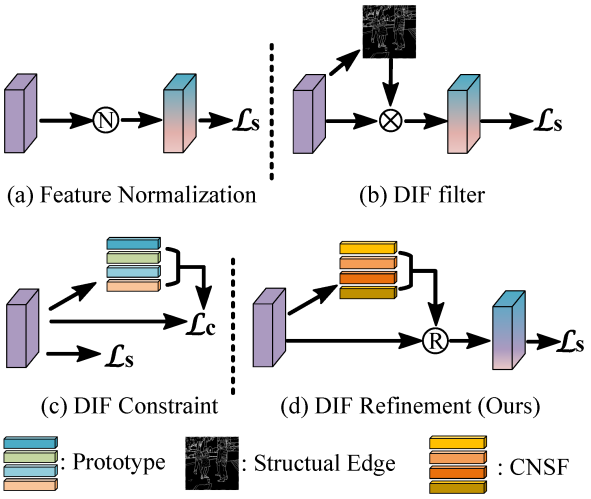

Figure 2: Different ways to utilize domain-invariant features (DIFs). Ⓝ, ⊗, and Ⓡ refer to normalization, element-wise multiplication, and refinement, respectively. CNSF, a type of DIF, represents the proposed class-wise non-salient feature.

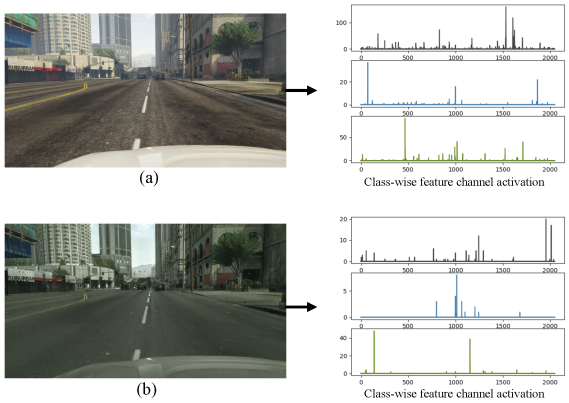

Figure 3: The class-wise feature channel activation in images with different styles ((a) for GTAV [14] style, (b) for Cityscapes [15] style), where the classes of road, sky, and vegetation are shown from top to bottom, respectively.

purpose of DG methods is to improve the robustness [11] and generalize it well to unseen domains. Several breakthroughs [12, 13] have recently been achieved in image generalized semantic segmentation (IGSS). However, the task of video generalized semantic segmentation (VGSS), which considers both temporal frames and domain generalizability, has not yet been explored. Compared to IGSS, the VGSS task is more in line with the dynamic attributes of the real world, which is significant for more accurate and robust predictions. Thus, with the aim of designing a learning framework for the VGSS task, we observe three critical phenomena.

1. **Domain-invariant features (DIFs) improve the generalizability of the model** because they remain invariant when the styles change. Obtaining and utilizing DIFs are the two main challenges in handling the domain shift problem. As shown in Figure 2, existing studies mainly focused on DIF selection and constraints. One type of DIF selection, as shown in Figure 2 (a), aims to process all deep features and normalize them, such as feature normalization and whitening [16, 17]. Another type of DIF selection, as shown in Figure 2 (b), forces neural networks to pay more attention to DIFs, such as structural edges [18] and features with small gradients [19]. Based on DIF constraints, some methods extract class prototypes (the centroid of the class feature) and perform distribution alignment between the class feature and class prototype [20, 21], as shown in Figure 2 (c). These studies have demonstrated that DIFs improve the model generalizability.
2. **The feature channel activations of the same classes in different domains have gaps.** Figure 3 shows the class-wise feature channel activations of the last layer in two images with the same content but different styles. Three classes are depicted for simplicity. A large gap between the diverse domains in the channels means that the model can perceive style information from the training images in addition to semantic information. Style information affects the class channel distribution, which may result in misclassification.
3. **Prediction inconsistency between adjacent temporal frames degrades the generalization and accuracy.** Almost all video tasks in computer vision suffer from temporal inconsistency problems [6, 22], which may lead to performance deterioration. Previous VSS studies alleviated this issue under the condition of independent and identically distributed (i.i.d.) data but they may not be effectively applied to the VGSS task. More recently, TCR [23] and TPS [24] were proposed for temporal adversarial consistency and temporal pseudo consistency for video domain adaptation in semantic segmentation, respectively, and were demonstrated to assist in the domain-invariant representation extraction. Thus, temporal consistency should be considered in the VGSS task for better learning of the domain invariant representations.

Thus, to improve model generalizability by reducing class misclassification and inter-frame inconsistency, three key points should be considered. (1). What kind of feature can be used to effectively represent DIFs? (2). Is there an approach that can refine the feature channels to select and enhance generalized channels adaptively? (3). How to reduce temporal inconsistency beneficially in the VGSS task? Based on the above, we devise a VGSS framework known as Non-Salient Feature Reasoning and Consistency (NSFRC), with its fundamental concept depicted in Figure 1. Specifically, we first define the class-wise non-salient feature, which describes the features of the class-wise non-salient region that carry more generalizable information, and can be considered as a type of DIF. We then propose a class-wise non-salient feature reasoning strategy to select

Table 1
Task setting comparison. ISS and VSS represent image semantic segmentation and video semantic segmentation, respectively.

Task	Continuous frames	Domain gap	Access target distribution
Image semantic segmentation			✓
Video semantic segmentation	✓		✓
Unsupervised domain adaptation for ISS		✓	✓
Unsupervised domain adaptation for VSS	✓	✓	✓
Image generalized semantic segmentation		✓	
Video generalized semantic segmentation (Ours)	✓	✓	

and enhance generalized channels using the class-wise non-salient features and graph relationship reasoning. In contrast to approaches shown in Figure 2 (a), (b), and (c), we explore a novel form (see Figure 2 (d) known as FID refinement, which considers DIFs as prior information to attend to the training process and assists the model in adaptively refining the features. Finally, we propose an inter-frame non-salient centroid alignment loss to reduce the gap between class-wise non-salient centroids of two adjacent frames. The main contributions of this work could be concluded as follows.

- We explore a new task known as VGSS, to handle dynamic scenes in real-world environments. To the best of our knowledge, this has not been studied in existing research. In addition, we propose the corresponding VGSS framework NSFRC.
- Based on the novel observation that channel activations between diverse domains have a discrepancy, we propose a class-wise non-salient feature reasoning strategy to select and enhance generalized channels adaptively, which also provides a new form for utilizing domain-invariant features.
- We propose the inter-frame non-salient centroid alignment loss to deal with the temporal inconsistency problem and alleviate negative transfer.
- We generalize our NSFRC framework from the VGSS task to the IGSS task, and achieve competitive results compared with its counterparts in multiple challenging benchmarks for both tasks.

Remainder of this paper is organized as follows. Section 2 discusses the related research fields. Section 3 introduces the NSFRC framework. Section 4 presents the extensive experiments and an analysis of the results. Section 6 concludes the paper.

2. Literature review

In this section, we first review several tasks including image semantic segmentation, video semantic segmentation, unsupervised domain adaptation, and domain generalization. The relationships among these are shown in Table 1. Thereafter, some related works on class activation map, prototypical alignment, and graph convolution are introduced.

2.1. Image semantic segmentation

Semantic segmentation is a typical computer vision task that predicts the semantic classes of each pixel in an image [25, 26]. Semantic segmentation methods can be roughly divided into architectural design and richer context aggregation methods. For architectural design, a fully convolutional network [27] and U-Net [28] have been used as the baselines of many existing sophisticated methods as they maintain both coarse and refined information depending on the skip connection operation. HRNet [29] maintains a semantically strong high-resolution feature map. In recent years, SETR [30] and SegFormer [31] have been developed as Transformer-based architectures that convert the original segmentation into a sequence-to-sequence prediction task. RepMLPNet [32] is a multilayer perceptron block with three fully connected layers to capture local priors via locality injection. Some methods obtain richer contexts using multi-scale information fusion, such as DeepLab-series methods [33, 34] and PSPNet [35]. Attention is also commonly used for capturing long-range dependencies; for example, in CC-Net [36], Non-local [37], and SegNeXt [38].

2.2. VSS

VSS has received widespread attention because it further considers the dynamic attributes of the real world. These methods can be mainly divided into reducing the cost of per-frame computation and improving the segmentation performance. To reduce the cost of per-frame computation, DFF [39] calculates the optical flow between the keyframe and current frame and obtains the predicted results by the wrapping operation. DVS [40] is a dynamic selection strategy that dynamically adopts a keyframe or segmentation network for semantic segmentation. LLVSS [41] obtains predicted results by fusing the low-level features of the current frame and high-level features of the keyframe. To devise a video framework at a low price, DAFC [42] was proposed as a distortion-aware feature correction method for correcting features in distorted regions, while preserving the propagated features for other regions. Inter-frame fusion and consistency are two core methods for devising a superior VSS framework to improve the segmentation performance. The keyframe usually serves as the prior information to refine the features of the current frame. EFC [43] jointly learns the video segmentation and optical flow tasks to improve both. GCS [44] is a guided co-segmentation network that simultaneously incorporates the short, middle, and long-term temporal inter-frame relationships. STT [6]

used Transformer-based architecture to balance accuracy and efficiency. TMA [7] includes inter-frame self-attention to perceive the inter-frame relationships better. In addition, the temporal consistency constraint ensures the prediction consistency of the temporal frames. UTC [45] provides an unsupervised temporal consistency loss to penalize unstable segmentation results. EFC [43] achieves temporal consistency outside the occluded regions to reduce the effects of unstable occluded regions. However, these methods cannot generalize effectively to other unseen domains because of the distribution gap between the source domain (domain for training) and unseen target domain (domain for testing).

2.3. UDA

UDA aims to perform effectively in the target domain, given the label source and unlabeled target domains [46]. Domain distribution alignment, self-training strategies, and sample mixing strategies are the three general methods for UDA semantic segmentation. Domain distribution alignment involves image-level [47, 48], feature-level [49, 50], and output-level [51–53] distribution alignment. The self-training strategy supervises unlabeled target data using pseudo-labels. BDL [54] uses the maximum probability threshold to filter target pixels with a confident prediction. For sample mixing, a context-aware mixup architecture [55] has been employed to explore and leverage the context relationship between two domains. ProDA [56] uses class prototypes to rectify the pseudo-labels. More recently, TCR [23] and TPS [24] were proposed for temporal adversarial consistency and temporal pseudo consistency in video domain adaptation for semantic segmentation, thereby facilitating the extraction of domain invariant representations. However, the testing environment is unseen and varies in many practical applications, which is a limitation of UDA technology.

2.4. DG

DG targets generalize well to other unseen domains using only the labeled source domain, where there is a gap between the source and unseen domains [57]. Data generation aims to extend the data as much as possible to cover unseen domains. DPRC [58] generates synthetic images using the styles of auxiliary data by leveraging CycleGAN [59]. FSDR [60] randomizes images using different domain-invariant frequencies. GLTR [61] harmonizes the global and local texture randomizations. To extract domain-invariant features, as shown in Figure 2 (a), some methods, such as IBN-Net [16], SW [62], ISW [17], and SAN [12], perform normalization or whitening on all features to reduce domain-specific information. PinMem [13] was recently proposed as a meta-learning framework that memorizes domain-agnostic and class-wise distinct information to reduce the representation ambiguity. These DG methods only segment using a single image and may not achieve better performance in a dynamic scene owing to the lack of continuous frame information.

2.5. Class activation map

A class activation map (CAM) is used to identify the discriminative region [63–65] using a single forward pass. The visual interpretability of CAM can build trustworthy intelligent systems, CAM methods, such as Grad-CAM [64], and Grad-CAM++ [65] have been extensively explored. Instead of improving the CAM, CAM technology is also widely used in the weakly-supervised semantic segmentation [66, 67], which fully leverages the discriminative localization ability of CAM. More recently, CDG [68] was used to calculate the CAM of a model trained on other domains as the weight to determine the feature dropout in DG training. Similar to the above methods, we also leverage the discriminative localization ability of the CAM to identify the less discriminative region (i.e., the non-salient region) to obtain a generalized representation.

2.6. Prototypical alignment

The prototype constraint shown in Figure 2 (c) is an effective strategy and is used in many UDA methods, where the class prototype is the class feature centroid and can be regarded as a type of DIF. ProCA [69] includes a prototypical contrast adaptation that pulls closer to the pixel and its corresponding class prototype. BiSMAP [21] utilizes source and target prototypes together to degrade hard-source samples. BAPA-Net [20] performs prototype alignment between the mixed and source images. Note that these methods employ all spatial features to generate and align prototypes. In contrast to these methods, our method considers credible features to generate a centroid and treats a non-salient centroid as prior information to enhance the domain-invariance model. Meanwhile, the proposed non-salient centroid alignment introduces temporal dimension alignment rather than spatial dimension alignment.

2.7. Graph convolution

Graph reasoning has become a popular means of constructing graph relationships for graph analogs. It has recently been used extensively in the semantic segmentation task, which can be roughly grouped into three types: spatial, class-wise, and temporal graphs. In the first type, the graph is constructed in spatial features. DGCN [70] uses a coordinate graph and feature graph in the spatial dimension. CNN-G [71] constructs a spatial graph considering the distance-based and semantic relationships. SPGR [72] explores multi-scale spatial graphs to enhance long-range contextual capture. MDGCN [73] employs a superpixel-based graph to adapt to various object distributions and geometric appearances. In terms of class-wise graphs, ADD-GCN [74] models the relation of content-aware category representations as a graph. CD-GCN [75] adopts a coarse-to-fine paradigm to learn the feature aggregation and weight allocation. For temporal graphs, SST-GCN [76] uses a stacked hourglass architecture to enable accurate action boundaries. These previous works used the strong ability of the relationship capture of graph convolution and achieved impressive performance. In our case, owing to the observation that feature channel activations of the same classes in different domains

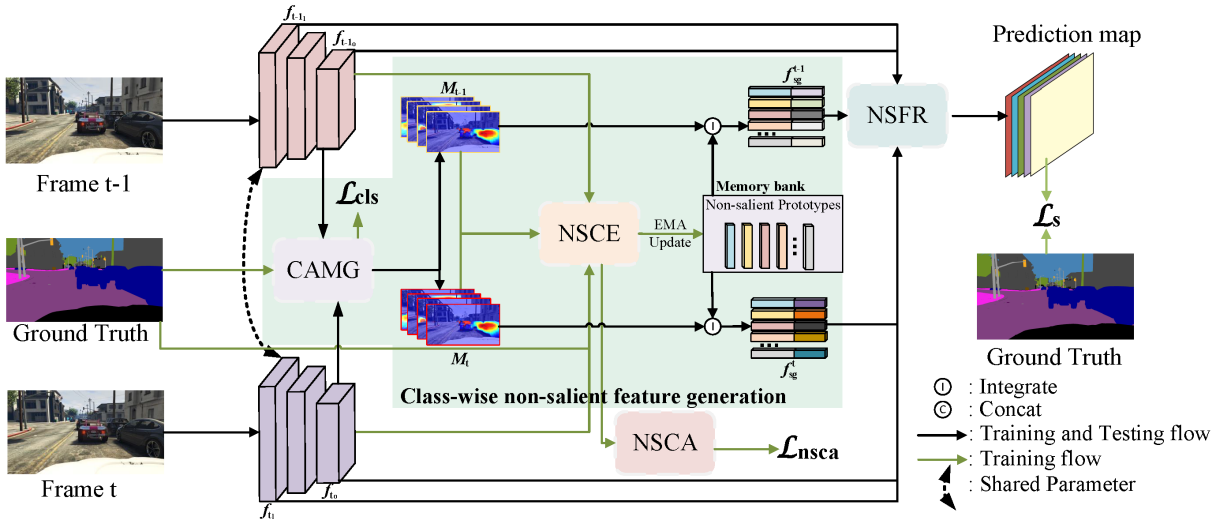

Figure 4: Overview of NSFRC framework. CAMG, NSCE, NSCA, and NSFR represent class activation map generation, non-salient centroid extraction, non-salient centroid alignment, and non-salient feature reasoning, respectively. Cubes represent deep features.

have a gap, we performed graph reasoning in the channel dimension to reduce class-wise confusion.

3. NSFRC framework

3.1. Problem statement

Given a seen source domain $\{X_s, Y_s\} \in D_s$ and K unseen target domains ($\{X_{t1}, Y_{t1}\} \in D_{t1}$, $\{X_{t2}, Y_{t2}\} \in D_{t2}$,...,$\{X_{tK}, Y_{tK}\} \in D_{tK}$), a DG model is trained using the source domain D_s and then evaluated on these K unseen target domains (D_{t1},...,D_{tK}), which aims to generalize well on all unseen target domains. X_* and Y_* are images and labels from different domains, respectively. The main difference between the IGSS and VGSS tasks is the input data. The input of the image-based DG model is one image x_*. However, the input of the VGSS task modifies one image x_* into two (x_*^{t-1}, x_*^t) or more temporally continuous frames (x_*^{t-n},..., x_*^{t-1}, x_*^t), where the previous frames ($x_*^{<t}$) are used as the prior information to refine the segmentation of current frame x_*^t.

3.2. Overview of framework

As illustrated in Figure 4, the proposed NSFRC framework contains class-wise non-salient feature generation, non-salient feature reasoning (NSFR), and non-salient centroid alignment (NSCA), where the class-wise non-salient feature generation is constructed using CAM generation and non-salient centroid extraction. The final objective of this framework is defined as:

$$\mathcal{L} = \mathcal{L}_s + \beta_1 \mathcal{L}_{cls} + \beta_2 \mathcal{L}_{nsca} \tag{1}$$

, where $\mathcal{L}_s$ is the segmentation loss using the cross-entropy function in Equation (15). $\mathcal{L}_{cls}$ and $\mathcal{L}_{nsca}$ represent the classification loss (Equation (4)) and non-salient centroid alignment loss (Equation (16)), respectively. β_1 and β_2 are hyperparameters of the loss terms.

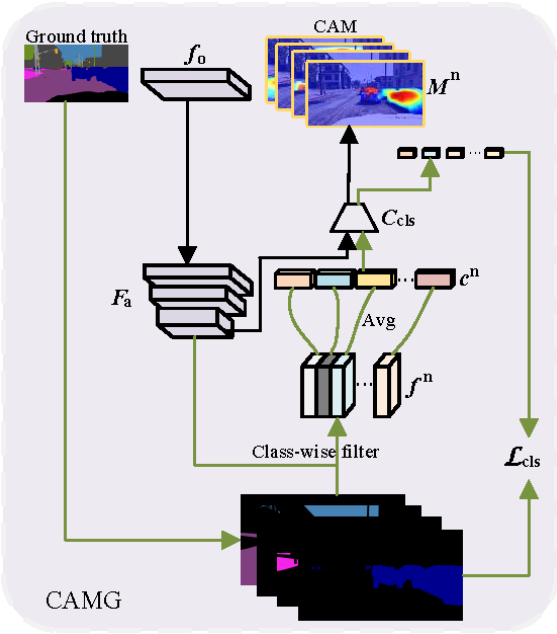

Figure 5: Depiction of CAM generation. At the training stage, the auxiliary classifier model updates parameters to learn meaningful CAM, while the auxiliary model captures CAM for the NSFR module at the test stage. Black lines denote the flow of both stages and green lines denote the flow of the training stage.

Our framework obeys two devising keys of video-based methods mentioned in the Literature Review section: inter-frame fusion and inter-frame consistency. To enhance the generalizability of the model fully, we further embed the consideration of DG into the above guidelines. As shown in Figure 4, class-wise non-salient features are obtained

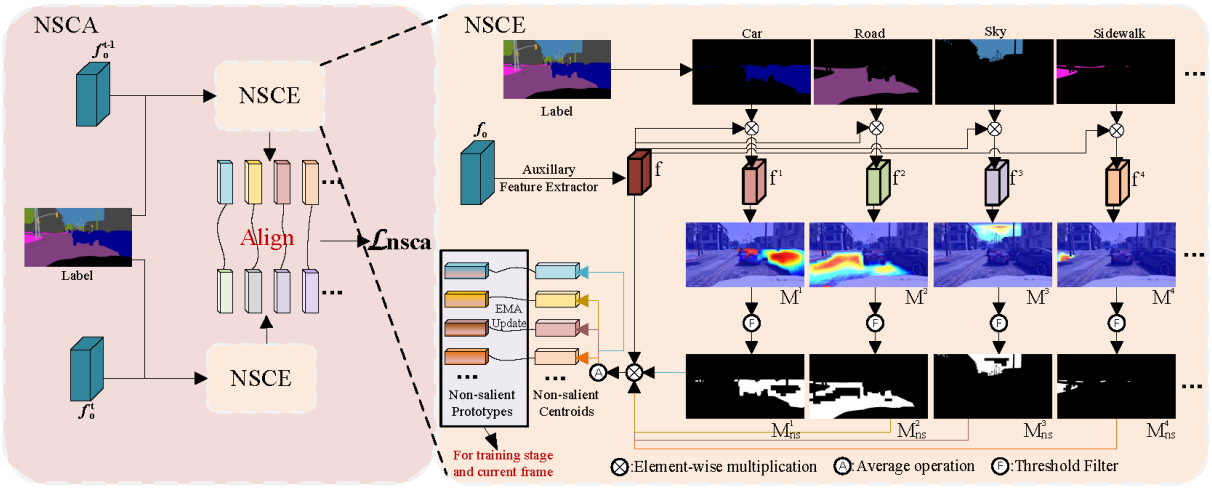

Figure 6: Structure of class-wise non-salient centroid extraction and non-salient centroid alignment. Note that non-salient prototypes are learned at the training stage and are sub-parts of class-wise non-salient features.

separately in the previous and current frames, which provide a more generalized representation for the following module. NSFR is performed for inter-frame fusion. The NSFR module adaptively selects generalized channels and enhances the features using class-wise non-salient features and graph relationship reasoning. For inter-frame consistency, the NSCA module is proposed to align the distribution of non-salient centroid between adjacent temporal frames to alleviate prediction inconsistency between the adjacent temporal frames.

3.3. Class-wise non-salient feature generation

Because domain-invariant features help to improve the model generalizability, as noted in Section 1, the class-wise non-salient feature with more domain-invariant information is defined, which is formed by the non-salient prototype and CAM. The steps of the class-wise non-salient feature generation include CAM generation, non-salient prototype extraction, and class-wise non-salient feature integration.

3.3.1. CAM generation

To obtain a CAM, a classification network is introduced as an auxiliary task. Given the deep feature f_o extracted by the segmentation feature extractor, f_o is pulled into the feature extractor of the auxiliary classifier $F_a(\cdot)$. Then, the $n \in N$ class feature f^n is filtered by the one-hot encoded ground truth y_s, which is denoted as:

$$f^n = F_a(f_o)\mathbb{1}(y_s^{(h,w,n)} == 1) \tag{2}$$

, where $\mathbb{1}$ is the indicator function, $\mathbb{1}(\cdot) = \begin{cases} 1, y_s^{h,w,n} == 1 \\ 0, y_s^{h,w,n} == 0 \end{cases}$. The class n feature centroid c^n is obtained by the average of the class n feature f^n, which can be defined as:

$$c^n = \frac{\sum_{x_s \in X_s} \sum_h \sum_w f^n}{\sum_{x_s \in X_s} \sum_h \sum_w \mathbb{1}(y_s^{(h,w,n)} == 1)} \tag{3}$$

Thereafter, c^n is pulled into the auxiliary classifier C_{cls}^n to obtain the prediction. The classifier loss $\mathcal{L}_{cls}$ is calculated using the cross-entropy:

$$\mathcal{L}_{cls} = -\sum_{i=0}^{N} y^n log(C_{cls}^n(c^n)) \tag{4}$$

, where $\mathcal{L}_{cls}$ is designed to capture meaningful weights of the classifier. Finally, the class n activation map M^n is denoted as:

$$M^n = \sum_{g=1}^{G} w_g^n f_g^n \tag{5}$$

, where $g \in G$ is the channel of the feature. w_g^n represents the weight of class n in channel g, which belongs to the auxiliary classifier C_{cls}^n and is learned by Equation (4). The process of this module is shown in Figure 5.

3.3.2. Non-salient prototype extraction

As indicated by Maxdrop [77] and RSC [19], most predictive parts contain less domain-invariant information. That is, the non-salient region is expected to be a more generalized region. The non-salient region in the feature map is identified by the CAM [63] owing to its discriminative localization ability. The class n non-salient mask M_{ns}^n is obtained by a threshold filter, which is denoted as:

$$M_{ns}^n = \begin{cases} 1 & 0 < M^{n(h,w)} <= M^n(\alpha) \\ 0 & M^{n(h,w)} > M^n(\alpha) \end{cases} \tag{6}$$

, where α is a hyperparameter to represent the pixel percentage that needs to be filtered. $M^n(\alpha)$ is the J^{th}-largest value in M^n, where $J = H \times W \times \alpha$. The non-salient centroid p^n is calculated as the average value of f^n under non-salient mask M_{ns}^n, which is denoted as:

$$p^n = \frac{\sum_{x_s \in X_s} \sum_h \sum_w f^n \mathbb{1}(M_{ns}^{n(h,w,n)} == 1)}{\sum_{x_s \in X_s} \sum_h \sum_w \mathbb{1}(M_{ns}^{n(h,w,n)} == 1)} \tag{7}$$

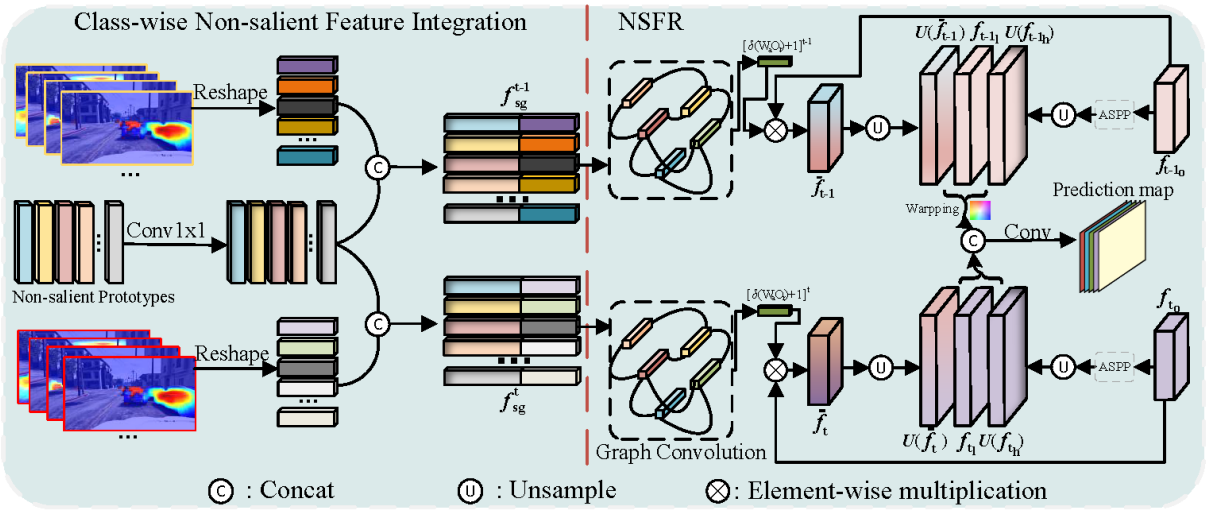

Figure 7: Depiction of class-wise non-salient feature integration and non-salient feature reasoning.

, where p^n is also the initial value of the non-salient prototype. An exponential moving average (EMA) operation between the two iterations is used to update the non-salient prototype, which can be represented as:

$$p^n \leftarrow \lambda p^n + (1-\lambda)p'^n \tag{8}$$

, where p'^n represents the non-salient centroid using the current frame in Equation (7). The non-salient prototype p^n is more generalizable compared to the naive prototype obtained using Equation (3) because the less generalized representation is filtered. For clarity, Figure 6 exhibits the extraction of the non-salient centroid and non-salient prototype. During the training stage, the non-salient prototype undergoes updates at every iteration, and the parameters of the prototypes at the final iteration are saved. In the testing stage, the non-salient prototype employs the stored prototype parameters and maintains them as fixed.

3.3.3. Class-wise non-salient feature integration

As indicated in the above section, the value of the CAM is linked to the degree of generalization of the spatial pixel, where the non-salient region carries more generalized information to achieve domain generalization. Thus, the CAM perceives the generalized importance of spatial features, which is employed as significant information constructing class-wise non-salient features. Specifically, the class-wise non-salient feature f_{sg}^n is defined as the concatenation of the non-salient prototype p^n and CAM M^n, which is denoted as:

$$f_{sg}^n = \Psi(Conv(p^n), Flatten(M^n)) \tag{9}$$

, where $\Psi(\cdot)$ and $Flatten(\cdot)$ refer to feature concatenation and feature flattening, respectively. Figure 7 depicts integration operation process.

3.4. Class-wise NSFR

Another concern is the use of class-wise non-salient features. As explained in Section 1, the feature channel activations for the same class in different domains have a gap, which means that the model can perceive domain information from training images rather than only the semantic information. This motivated us to propose a channel distribution reweighting strategy for adaptively selecting and enhancing the generalized channels of the original features. Furthermore, previous studies have shown that DIFs can enhance the model generalizability, as shown in Figure 2. Combining these two observations, a straightforward concept is to embed the DIFs into the original features to assist in reducing the domain-variant information of the original features, which is known as DIF refinement. In particular, we propose class-wise non-salient feature reasoning to achieve DIF refinement. Class-wise non-salient features, as a type of DIFs, are adopted as the input for feature reasoning, which captures the inter-class relationship of the DIFs to select generalized channels and enhance the features adaptively.

As graph convolution excels in capturing node relationships and adaptively propagating information [78, 79], it is suitable for adoption as a class-wise relationship reasoning method. Given a graph $\mathcal{G}$ containing nodes $\mathcal{N}$ and edges $\mathcal{E}$, the graph convolution can be defined as:

$$O_r = \sigma(W_r f_{sg} A_r) \tag{10}$$

, where O_r, A_r, and W_r are the output, adjacency matrix (i.e., the relationship between nodes), and learnable weight matrix, respectively. $\sigma(\cdot)$ denotes the non-linear activation function. To capture the relationship between different classes, the non-salient features of each category are employed as nodes in the graph. A 1×1 convolution layer is used to get the adjacent matrix A_r as in GloRe [78]. Meanwhile, the relationship reasoning also conducts Laplacian matrix smoothing $(I - A_r)$ using the a residual sum between the

identity matrix I and adjacent matrix A_r to propagate the node features over the graph more effectively. Therefore, the graph relationship reasoning can be rewritten as:

$$O_r = \sigma(W_r f_{sg}(I - A_r)) \tag{11}$$

Thereafter, another 1×1 convolution W_a is used to match the channel dimension of original feature f_o. The reconstructed feature $\bar{f}$ can be defined as:

$$\bar{f} = \delta(W_a O_r) * f_o + f_o \tag{12}$$

, where $\delta(\cdot)$ is the sigmoid operation. Note that the shape of the reasoned feature $\delta(W_a O_r)$ is $1 \times C_{f_o}$, where C_{f_o} is the channel number of f_o.

The effect of the proposed feature reasoning is twofold. First, the reconstructed feature is expected to learn and activate generalized and representative channels because the relationship between class-wise non-salient prototypes containing more domain-invariant information is captured and reasoned. Second, a new form that differs from that Figure 2 (a), (b), and (c) is adopted; that is, DIFs are embedded into the original features to reduce the domain-variant information of the original feature, which potentially provides valuable insights and inspiration for follow-up studies. It can be seen that the proposed method compiles well with the first and second observations mentioned in Section 1.

Temporal feature fusion between two frames is employed to integrate the segmentation results of the temporal frames for more accurate prediction. Considering the temporal frames, the final feature of the t^{th} frame f_t can be concatenated from the high-level feature f_{t_h}, low-level feature f_{t_l}, and reconstructed feature $\bar{f}_t$, which can be defined as:

$$f_t = \Psi(U(f_{t_h}), f_{t_l}, U(\bar{f}_t)) \tag{13}$$

, where the original feature f_{t_o} is pulled into the $ASPP$ module [33] to aggregate the multi-scale context, where $f_{t_h} = ASPP(f_{t_o})$. The low-level feature f_{t_l} is the feature of stage 1 in the backbone network. U refers to the upsample operation to match the dimension of f_{t_l}. The temporal fused prediction P_{fuse} is obtained by concatenating the predictions of two frames, which is denoted as:

$$P_{fuse} = C_{fuse}(\Psi(C(f_t), \mathcal{W}(C(f_{t-1}), \mathcal{F}))) \tag{14}$$

, where $\mathcal{W}$ is the warping operation, and $\mathcal{F}$ is the optical flow estimated by FlowNet-V2 [80]. C_{fuse} is the classifier for the temporal fused prediction. Finally, a cross-entropy function is leveraged as the segmentation loss:

$$\mathcal{L}_s = -\sum_{h,w}\sum_{n\in N} y_s^{(h,w,n)} log(P_{fuse}^{(h,w)})) \tag{15}$$

Figure 7 depicts the process of the NSFR module.

3.5. NSCA

Inspired by TCR [23] and TPS [24], which alleviate temporal inconsistency in video domain adaptation semantic segmentation for better learning of the domain-invariant representation, we propose the NSCA loss to constrain adjacent frames, as shown in Figure 6, which can be described as follows:

$$\mathcal{L}_{nsca} = \frac{1}{N}\sum_{n=0}^{N} |p_{t-1}^n - p_t^n| \tag{16}$$

where the non-salient centroids p_{t-1}^n and p_t^n represent the class centroids in the $(t-1)^{th}$ and t^{th} frames, respectively, calculated by Equation (7).

The effect of the alignment loss is two-fold. First, the inter-frame feature alignment alleviates the temporal inconsistency problem to learn the domain-invariant representation more effectively. Second, compared with the naive centroid alignment, as the non-salient centroid is generated by the more generalized region, this strategy encourages the alignment of generalized features between different frames rather than the global features. This learning strategy filters out the less generalized features (i.e., unrelated source features) to alleviate the effect of the outliers in the global features; that is, it alleviates negative transfer [81].

4. Experiments

Extensive experiments were conducted to verify the superiority of our NSFRC framework in the VGSS task, including qualitative and quantitative comparisons and ablation studies. Our NSFRC framework was also extended to demonstrate its effectiveness in the IGSS task.

4.1. Dataset

Nine datasets were used in both the IGSS and VGSS tasks, including four real-world datasets (Cityscapes [15], CamVid [83], Mapillary [84], and BDD100K [85]) and five synthetic datasets (VIPER [86], GTAV [14], SYNTHIA [87], V2X [88], and VKitti2 [89]).

Four synthetic datasets and two real-world datasets were used in the VGSS task. The VIPER dataset is a synthetic dataset of urban scenes containing over 25000 video frames with FHD resolution (1920×1080) under different environmental conditions, which were captured in the computer game Grand Theft Auto V. The SYNTHIA-Seq dataset, which is a sub-dataset of SYNTHIA, has 8500 video frames with eight views, where the images of six views and the other two views are used as the training and validation sets, respectively. The large-scale V2X dataset contains 37330 video frames with a resolution of 1600×900 for autonomous driving, supporting multi-agent multi-modality research. Four sequences (35030 frames) and one sequence (2300 frames) are used as the training and the testing sets, respectively. The Vkitti2 dataset is a large-scale dataset containing five scenes (42520 video frames) with a resolution of 1242×375. Similar to the V2X dataset, four scenes (33580 frames) and one scene (8940 frames) in the VKitti2 dataset are used as the training and the validation sets, respectively. For the real-world datasets, 5950 training and 1000 validation video frames with a resolution of 2048×1024 were adopted from

Table 2
Quantitative comparisons for the VGSS task on widely-used datasets. The best and second-best performances are represented by **bold** and underline, respectively. → refers to "generalize to". Avg refers to the average mIoU on different evaluation datasets. DG and Video represent domain generalization strategy and continuous frames, respectively. Model T, R, M, and S denote Transformer, ResNet-50, MobileNet-V2, and ShuffleNet-V2, respectively.

Methods	Model	Avg	DG	Video	VIPER (V_s)→					Synthia-Seq (S_s)→					CamVid (CV_s)→					Cityscapes-Seq (C_s)→				
					→S_s	→C_s	→CV_s	→$V2_s$	→VK_s	→V_s	→C_s	→CV_s	→$V2_s$	→VK_s	→V_s	→C_s	→S_s	→$V2_s$	→VK_s	→V_s	→CV_s	→S_s	→$V2_s$	→VK_s
CFFM [82]	T	30.4		✓	34.2	26.8	28.3	33.0	26.9	18.2	29.8	37.4	27.9	27.4	18.5	39.1	18.7	24.7	25.8	24.9	46.0	41.1	38.3	40.1
TMA [7]	R	29.5		✓	35.4	30.9	34.9	38.3	31.4	11.5	12.0	19.4	10.7	20.9	18.9	44.5	26.4	35.6	29.3	24.0	52.5	33.0	44.6	36.1
IBN [16]	R	36.4	✓		36.3	29.6	33.5	30.5	35.2	21.1	40.0	45.7	29.9	37.4	21.6	44.6	39.6	29.6	37.3	23.0	60.2	40.9	42.7	49.6
SW [62]	R	39.1	✓		**45.2**	28.9	35.2	34.3	40.3	22.6	41.0	45.7	**37.6**	**42.0**	22.9	47.7	40.2	37.6	**38.5**	21.9	60.8	43.8	45.5	**50.9**
ISW [17]	R	37.7	✓		42.3	34.0	38.6	40.7	40.6	24.4	40.1	44.6	30.7	39.1	21.1	43.7	39.2	30.5	34.9	20.8	60.1	40.5	39.1	48.1
PinMem [13]	R	38.9	✓		40.6	30.8	**43.2**	43.9	33.2	21.4	37.5	32.1	35.9	38.9	23.9	**56.0**	43.6	36.4	37.5	29.9	61.1	40.2	43.0	48.9
NSFRC (Ours)	R	**42.2**	✓	✓	41.1	**36.0**	39.6	**45.9**	**40.9**	**26.1**	**42.1**	**47.8**	**37.6**	41.1	**27.2**	52.7	**43.7**	**41.4**	37.9	**33.1**	**61.7**	**50.9**	**48.9**	47.6
TMA [7]	M	28.7		✓	31.6	26.7	36.3	30.9	24.4	18.5	15.7	23.0	23.2	23.6	20.8	44.1	36.9	37.2	29.8	18.9	43.2	20.1	39.4	29.1
IBN [16]	M	34.6	✓		35.8	26.2	34.2	30.1	33.1	21.6	35.9	40.0	27.6	37.2	18.5	43.9	37.4	**41.4**	35.7	17.4	55.5	36.8	37.0	46.2
SW [62]	M	36.1	✓		38.3	23.5	37.5	41.3	29.7	20.0	34.7	42.6	31.2	38.8	21.4	46.2	**40.4**	39.6	**37.0**	15.3	57.7	34.6	**47.7**	44.0
ISW [17]	M	35.3	✓		35.5	25.6	34.6	28.9	34.8	21.6	35.9	40.8	30.7	**39.5**	18.7	43.7	36.3	40.2	34.6	19.2	**58.6**	39.1	41.2	47.1
PinMem [13]	M	35.2	✓		37.6	27.1	**43.3**	33.3	31.8	18.7	24.9	31.0	24.5	37.6	23.8	**51.5**	36.2	31.9	35.3	29.4	56.0	43.9	38.5	**48.1**
NSFRC (Ours)	M	**39.3**	✓	✓	**38.8**	**31.9**	34.3	**41.9**	**36.7**	**26.2**	**39.8**	**45.7**	**37.4**	39.0	**25.1**	50.1	39.1	39.5	36.7	**30.4**	55.3	**45.6**	45.3	46.5
TMA [7]	S	28.3		✓	33.6	26.9	30.5	34.5	27.5	13.6	7.5	22.8	17.7	21.9	21.3	43.3	36.1	37.3	31.4	19.9	45.7	27.1	37.7	29.9
IBN [16]	S	32.4	✓		34.5	23.3	27.9	29.0	37.2	23.3	32.5	43.1	26.8	36.6	19.5	38.1	31.1	28.2	32.2	16.9	54.6	35.5	33.4	44.7
SW [62]	S	32.5	✓		35.4	21.8	33.6	28.2	**37.8**	21.7	33.5	40.8	25.2	32.1	19.9	39.2	32.9	33.4	34.1	15.2	53.5	32.1	33.6	**46.0**
ISW [17]	S	32.3	✓		35.1	24.4	28.9	28.1	35.6	22.1	32.1	43.6	24.8	33.2	18.9	38.1	30.9	25.8	32.9	22.2	55.5	37.9	31.2	43.6
PinMem [13]	S	33.3	✓		32.3	**28.5**	29.8	34.9	26.5	17.2	23.7	26.7	25.8	34.6	22.7	**52.1**	**37.2**	33.0	34.8	27.4	**56.5**	42.0	35.3	44.6
NSFRC (Ours)	S	**37.6**	✓	✓	**36.6**	**28.5**	**36.3**	**42.5**	33.7	**23.8**	**35.2**	**44.8**	**35.4**	**37.5**	**24.2**	46.6	36.8	**38.5**	**35.0**	**27.9**	56.3	**42.5**	**45.3**	45.6

Table 3
Quantitative comparisons for the VGSS task on large-scale datasets. The best and second-best performances are represented by **bold** and underline, respectively. → refers to "generalize to". Avg refers to the average mIoU on different evaluation datasets. Model T, R, M, and S denote Transformer, ResNet-50, MobileNet-V2, and ShuffleNet-V2, respectively.

Methods	Model	Avg	V2X ($V2_s$)→					VKITTI2 (VK_s)→				
			→S_s	→C_s	→CV_s	→V_s	→VK_s	→S_s	→C_s	→CV_s	→V_s	→$V2_s$
CFFM [82]	T	23.1	17.3	25.1	16.6	16.6	19.6	**27.4**	32.6	**32.2**	17.2	26.6
TMA [7]	R	24.0	15.3	34.4	27.1	21.7	30.6	17.5	25.7	20.6	15.6	31.2
IBN [16]	R	27.4	27.7	30.8	28.1	20.7	30.3	22.3	36.0	27.4	17.3	33.1
SW [62]	R	29.7	32.8	35.6	29.9	24.1	32.3	21.9	36.5	28.7	**21.8**	33.7
ISW [17]	R	27.4	31.2	35.7	28.3	23.8	32.4	15.7	35.3	27.2	15.0	29.8
PinMem [13]	R	29.9	28.7	35.0	35.4	25.5	35.6	24.4	**37.1**	**32.2**	20.0	25.3
NSFRC (Ours)	R	**34.8**	**35.3**	**40.4**	**38.9**	**30.2**	**39.8**	25.4	37.0	32.0	20.3	**38.0**
TMA [7]	M	20.0	9.3	17.7	26.4	19.6	25.1	16.0	21.7	22.8	16.2	25.4
IBN [16]	M	27.1	27.7	31.1	29.1	18.5	28.5	**24.0**	**34.9**	27.7	17.6	31.9
SW [62]	M	27.6	21.5	33.8	31.1	20.2	29.5	21.0	33.3	28.9	**21.0**	**35.2**
ISW [17]	M	27.5	28.7	32.0	29.0	19.9	31.3	22.2	33.2	28.1	20.2	30.4
PinMem [13]	M	27.4	25.3	35.4	33.9	25.0	30.9	20.8	33.9	31.1	17.1	20.6
NSFRC (Ours)	M	**32.0**	**33.7**	**38.0**	**40.2**	**27.7**	**37.7**	23.9	32.1	**32.7**	18.8	35.0
TMA [7]	S	23.9	17.9	20.0	25.6	18.4	26.3	23.7	29.0	25.3	**20.4**	32.3
IBN [16]	S	25.7	23.5	29.3	29.2	17.6	29.0	20.9	32.8	28.3	19.6	27.0
SW [62]	S	26.9	24.2	29.8	28.3	19.9	31.3	**27.8**	29.3	**31.4**	18.0	28.9
ISW [17]	S	25.0	23.2	29.4	29.8	18.0	28.8	18.6	**33.6**	26.2	15.8	26.8
PinMem [13]	S	26.5	27.2	**32.5**	31.5	22.5	30.0	21.9	31.5	28.8	16.8	22.0
NSFRC (Ours)	S	**29.8**	**31.1**	32.0	**37.8**	**25.5**	**32.9**	23.6	32.9	30.2	18.7	**33.4**

the Cityscapes-Seq dataset. The CamVid dataset contains 366 and 101 video frames with a resolution of 960 × 720 for the training and validation sets, respectively.

Two synthetic and three real-world datasets were used in the IGSS task. For the synthetic datasets, the GTAV dataset collects over 25000 images with a high resolution of 1914 × 1052. The SYNTHIA-Rand set is the subset of the SYNTHIA dataset and consists of 9400 1280 × 760 images with different views. For real-world datasets, the BDD100K dataset is a large driving dataset, where 7000 training and 1000 validation images of 1280 × 720 resolution are used in the IGSS task. The Mapillary dataset covers six continents and consists of 18000 training and 2000 validation images, where the resolution of the images is at least FHD resolution. The Cityscapes dataset is widely used in the street understanding task, which contains 2975 training and 500 validation images with the resolution of 2048 × 1024.

4.2. Implementation detail

ResNet-50 [90], ShuffleNet-V2 [91], and MobileNet-V2 [92] were adopted as the backbone networks and the model pre-trained on ImageNet was used as the initial model. The segmentation model adopted a stochastic gradient descent optimizer based on the Pytorch library [17], where the initial learning rate, weight decay, and momentum are set to 0.01, 5e-4, and 0.9, respectively. The mean Intersection over union (MIoU) of the categories was reported as the performance metric for comparison. Similar to ISW [17], several photometric transformations, such as Gaussian blurring and color jittering, were used to avoid source overfitting. The hyperparameter of the filter for non-salient centroid generation was set to 0.3. For the VGSS task, all video frames were resized to 768×384. For the IGSS task, all input images were resized to 768×768 like other methods (ISW [17] and PinMem [13]). λ in the Equation (8) was set to 0.9 and the number of frames was 2. Both β_1 and β_2 were set to 0.001. The auxiliary feature extractor was built sequentially, consisting of a 3×3 convolution layer followed by batch normalization and ReLU activation, then another 3×3 convolution layer followed by batch normalization and ReLU activation. The auxiliary classifier used a 3×3 convolution layer.

4.3. Comparison with state-of-the-art methods

For the VGSS task, ResNet-50 [90], ShuffleNet-V2 [91], and MobileNet-V2 [92] were adopted as the backbone networks. The video datasets VIPER (V_s), Synthia-Seq

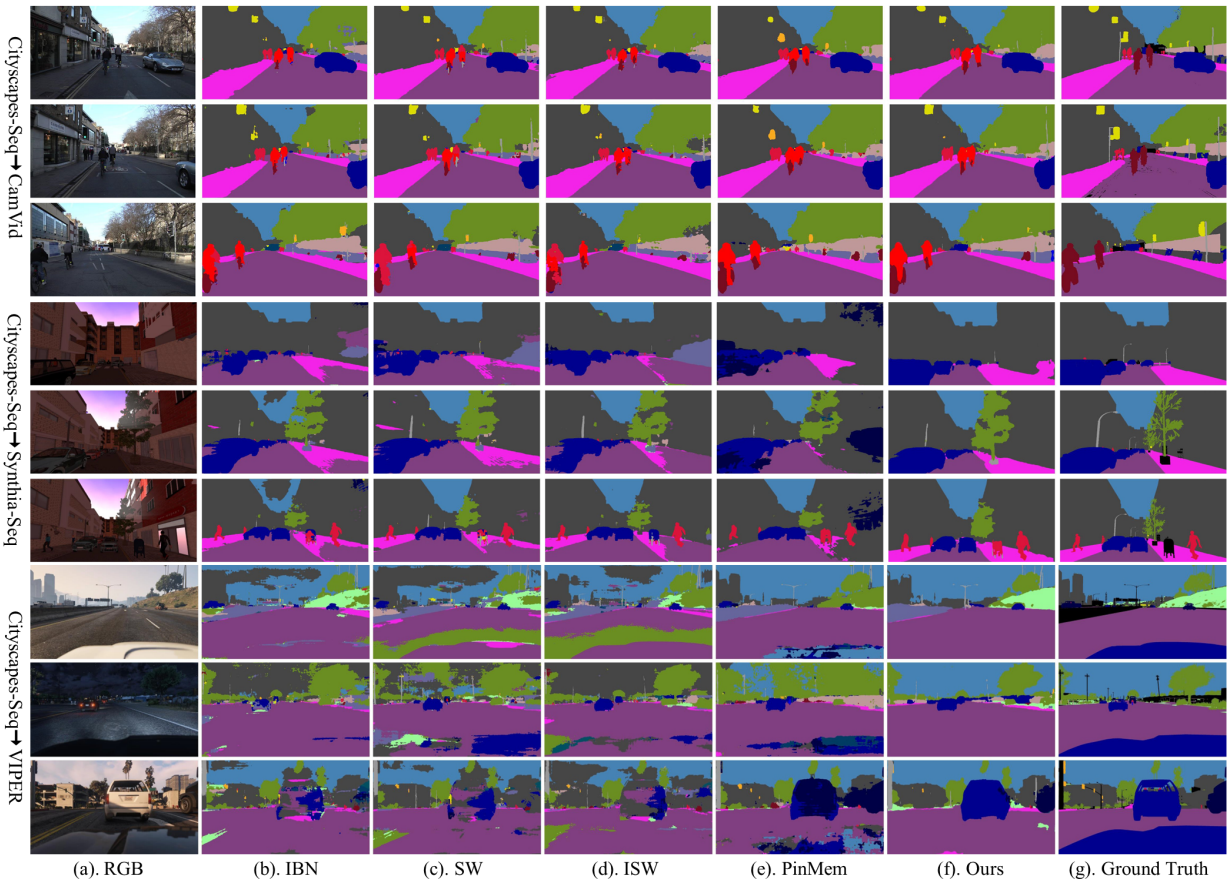

Figure 8: Result visualization comparison of state-of-the-art methods including IBN [16], SW [62], ISW [17], and PinMem [13]. Best viewed in color.

(S_s), CamVid (CV_s), Cityscapes-Seq (C_s), V2X ($V2_s$), and VKitti2 (VK_s) were used in turn as the source domain for training, while the other datasets were used as the test sets. Thus, the experiments consisted of $V_s \rightarrow \{S_s, C_s, CV_s, V2_s, VK_s\}$, $S_s \rightarrow \{V_s, C_s, CV_s, V2_s, VK_s\}$, $C_s \rightarrow \{V_s, CV_s, S_s, V2_s, VK_s\}$, $CV_s \rightarrow \{C_s, V_s, S_s, V2_s, VK_s\}$, $V2_s \rightarrow \{S_s, C_s, CV_s, V_s, VK_s\}$, and $VK_s \rightarrow \{S_s, C_s, CV_s, V_s, V2_s\}$. The left of the $\rightarrow$ is the source domain for training, whereas the right of the $\rightarrow$ is the target domain for evaluation. A single model was selected to evaluate all target domains instead of using multiple models such as DPRC [58]. For better readability, we divided the above experiments into two tables. Table 2 was used to evaluate the performance on widely-used datasets (VIPER, Synthia-Seq, CamVid, and Cityscapes-Seq), whereas Table 3 was employed to evaluate the performance on large-scale datasets (V2X and VKitti2).

Two evaluation metrics were employed to validate the effectiveness of our method. The first metric is the average mIoU in all evaluated settings for each method. A higher average mIoU reflects a better domain generalization ability. For example, for the Resent-50 backbone, we achieved the best performance with 42.2% in terms of the average mIoU in Table 2. However, if a method has good performance in only one domain while getting bad performance in other domains, it cannot meet competitive domain generalizability. From this viewpoint, the second metric is the number of best or second-best performances in different evaluated settings (as indicated in the rows in Table 2). A higher number of best or second-best performances in a method reflects that the generalization can be relatively evenly distributed across different domains. For example, for the Resnet-50 backbone, we achieved 14 best performances (including $V_s \rightarrow \{C_s, V2_s, VK_s\}$, $S_s \rightarrow \{V_s, C_s, CV_s, V2_s\}$, $CV_s \rightarrow \{V_s, S_s, V2_s\}$, and $C_s \rightarrow \{V_s, CV_s, S_s, V2_s\}$) and four second-best performances ($V_s \rightarrow \{CV_s\}$, $S_s \rightarrow \{VK_s\}$, and $CV_s \rightarrow \{C_s, VK_s\}$) in a total of 20 evaluated situations (V_s or S_s or CV_s or $C_s \rightarrow$ other five datasets). The performance comparison on widely-used datasets is presented in Table 2. In the ResNet-50 backbone, our method achieved 42.2% in terms of the average mIoU and 14 best results in 20 evaluation settings. A 3.1% improvement in the average mIoU shows the superiority of our approach compared to the second-best method. In the MobileNet-V2 backbone, our approach outperformed all the state-of-the-art methods with a significant improvement of at least 3.2% in average mIoU. In the ShuffleNet-V2 backbone, 15 best performances were

Table 4
Quantitative comparisons for the IGSS task on ResNet-50 backbone. The best and second-best performances are represented by **bold** and underline, respectively. → refers to "generalize to". Avg refers to the average mIoU on different evaluation datasets.

Methods	Model	Avg	GTAV (G)→				SYNTHIA (S)→				Cityscapes (C)→				BDD (B)→				Mapillary (M)→			
			→C	→B	→M	→S	→C	→B	→M	→G	→B	→M	→G	→S	→G	→S	→C	→M	→G	→S	→C	→B
IBN [16]	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 [62]	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 [58]	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 [61]	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 [17]	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 [12]	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 [13]	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
NSFRC (Ours)	ResNet-50	**42.2**	**42.6**	**37.9**	**42.0**	**33.1**	**39.5**	30.0	32.3	29.7	50.9	57.8	45.3	**30.5**	30.6	**33.6**	**57.4**	**56.2**	**49.7**	**34.6**	**60.0**	**51.2**

Table 5
Performance comparison in Foggy cityscapes and IDD datasets between the Baseline [17] and our method. → refers to "generalize to". Avg refers to the average mIoU on different evaluation datasets.

Methods	Avg	G→		S→		C→		B→		M→	
		→I	→F	→I	→F	→I	→F	→I	→F	→I	→F
Baseline	40.1	33.7	30.3	26.8	29.7	47.1	58.3	45.4	47.7	41.1	40.7
Ours	44.6	39.2	37.7	28.6	32.7	49.7	60.4	48.8	50.6	49.3	49.1

Table 6
Ablation studies on proposed component containing NSFR and NSCA. The model is trained on the Cityscapes-Seq with ResNet-50 backbone network.

Method	NSFR	NSCA	→ V_s	→CV_s	→S_s	Avg	Iter time(s)
Baseline			28.2	60.8	40.0	43.0	0.0107
NSCA		✓	32.6	57.4	50.0	46.7	0.0122
NSFR	✓		**34.1**	60.0	49.5	47.9	0.0123
NSFR + NSCA	✓	✓	33.1	**61.7**	**50.9**	**48.6**	0.0128

achieved in 20 evaluation settings. Compared to the second-best method, the result of our framework is improved to 37.6% and had a clear increase of 4.3% in terms of the average MIoU. These experiments and discussions indicate that the segmentation quality is enhanced by the proposed class-wise relationship reasoning and NSCA.

The performance comparison on large-scale datasets is presented in Table 3. In the Resnet-50, MobileNet-V2, and ShuffleNet-V2 backbones, our proposed method exhibited a clear increase of at least 4.9%, 4.4%, and 2.9% in terms of the average mIoU, respectively. Meanwhile, our method maintained relatively even generalizability to all testing domains, achieving the highest number of the best and second-best performances in all evaluated settings. For instance, our method achieved six best performances and four second-best performances in all 10 evaluated settings for the ResNet-50 backbone.

Two video-based segmentation methods, namely TMA [7] and CFFM [82] are performed on the same experiments to verify the importance of the DG strategy. As shown in Tables 2 and 3, TMA and CFFM always exhibited inferior performance to that of the DG methods, which shows that such video-based methods cannot handle environments with diverse styles. Thus, it is significant to fuse the DG and continuous frames.

We also provide a visual comparison with state-of-the-art methods including IBN [16], SW [62], ISW [17], and PinMem [13]. As shown in Figure 8, our methods achieved better segmentation results with more completed object shapes and fewer incorrect areas.

In addition, we report the performance in the IGSS task in Table 4, where the model was trained on Mapillary (M), GTAV (G), Cityscapes (C), BDD100K (B), and SYNTHIA (S) in turn similar to the VGSS task. Correspondingly, experiments with 20 evaluation settings were performed: $G \rightarrow \{C, B, M, S\}$, $S \rightarrow \{G, C, B, M\}$, $C \rightarrow \{G, S, B, M\}$, $B \rightarrow \{G, C, S, M\}$, $M \rightarrow \{G, S, C, B\}$. In the IGSS task, NSCA cannot perform alignment on inter-frame features because the input is a single image. Thus, NSCA was implemented by splitting the original features into two types of features by odd and even indices to calculate the non-salient centroid separately and perform centroid alignment. NSFR retained the original settings because this module does not require inter-frame information. The 13 best ($G \rightarrow \{B, M, C, S\}$, $S \rightarrow \{G\}$, $C \rightarrow \{S\}$, $B \rightarrow \{M, C, S\}$, and $M \rightarrow \{G, B, S, C\}$) and two second-best ($S \rightarrow \{C\}$ and $C \rightarrow \{B\}$) performances in our framework show that the NSFRC framework achieved state-of-the-art performance. Our approach improved the performance by 1.2% in terms of the average mIoU compared to the second-best method (i.e., PinMem [13]). In addition, we compared the performance of the baseline and our method on recent and challenging datasets (i.e., Foggy Cityscapes (F) [93] and IDD (I) [94]). As shown in Table 5, our method outperformed the baseline model with a 4.5% gain in the mIoU. These experiments verify that our proposed method is also effective for the IGSS task and outperforms other state-of-the-art methods.

4.4. Ablation study

4.4.1. Individual components

Ablation studies were conducted to verify the effectiveness of the proposed components. NSFR and NSCA were the ablation terms used for evaluation. Note that the model with only template feature fusion was the baseline; that is, it employed Equations (13), (14) without non-salient features. As shown in Table 6, the performance of the model trained on Cityscapes-Seq on the ResNet-50 backbone was reported and the remaining three datasets were evaluated. The baseline model denotes the model using only temporal

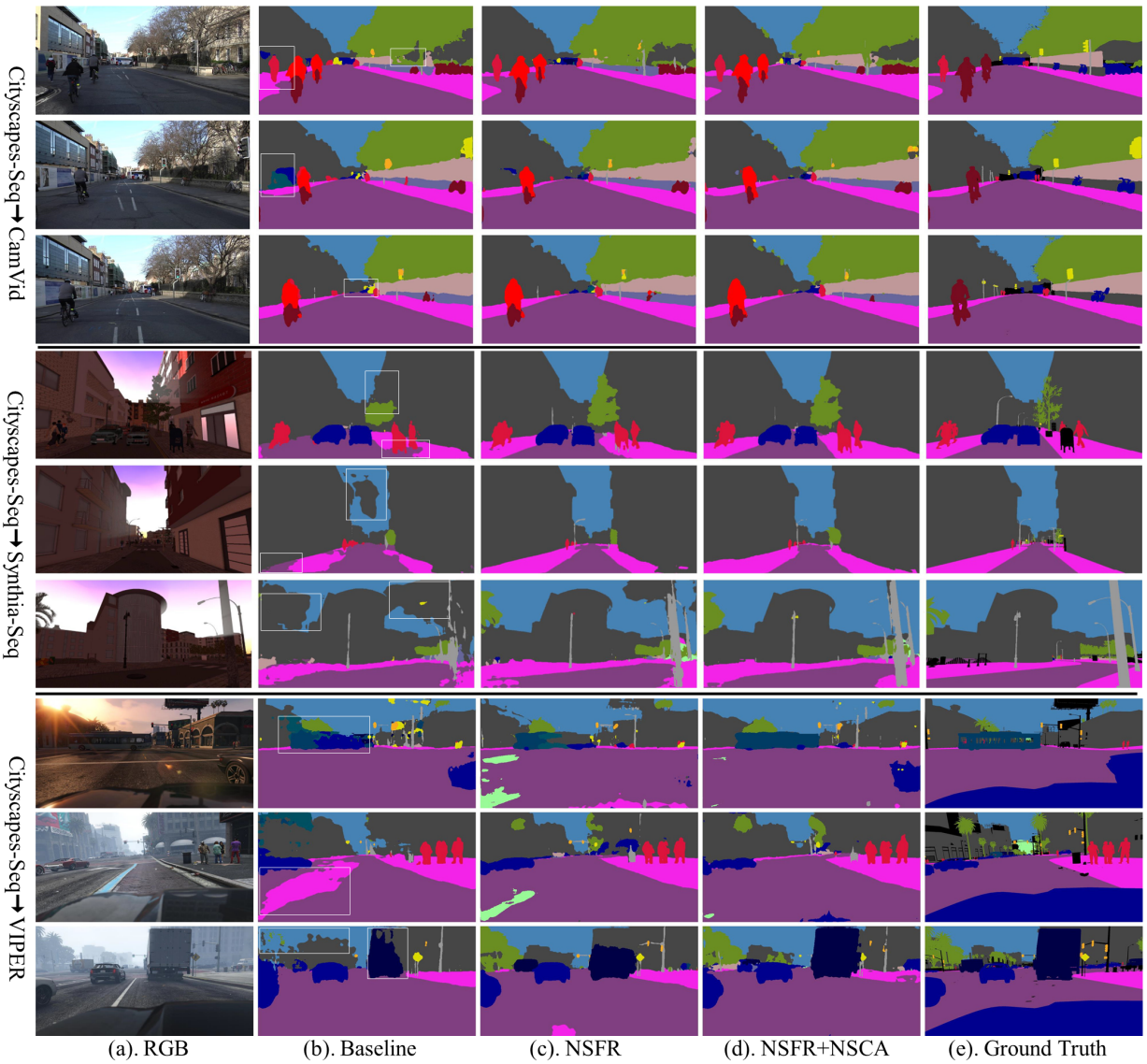

Figure 9: Result visualization comparison of different models on distinct domains. Best viewed in color. White boxes show the segmentation error in the baseline model.

feature fusion without the reconstructed feature $\bar{f}$ generated by non-salient feature reasoning. Our NSCA module achieved 46.7% in terms of the average mIoU with an 3.7% improvement compared to the baseline. Furthermore, NSFR achieved 47.9% in terms of the average mIoU. The performance of the final model was improved to 48.6% in terms of the average mIoU and the performance improvement was obvious compared to the baseline. Thus, the proposed modules contribute to enhancing the generalizability of the model. Meanwhile, there was no clear disparity in the iteration time among the proposed modules.

The segmentation visualizations of different models (the baseline model, model with NSFR, and model with NSFR + NSCA) on distinct domains ($C_s \rightarrow CV_s$, $C_s \rightarrow S_s$, $C_s \rightarrow V_s$) are provided for comparison in Figure 9. The NSFR model and final model exhibited better results than the baseline model and the final model showed smaller segmentation errors than the NSFR model, demonstrating that our framework alleviates the class confusion problem. For example, in the first row of $C_s \rightarrow V_s$, the bus shape in our final model was closer to the ground truth.

4.4.2. Non-salient region validation

In addition, to validate the effectiveness of the non-salient region, ablation studies were conducted on the sub-components of the proposed modules. First, as mentioned in Section 3.C, the class-wise non-salient feature was concatenated using the non-salient prototype and CAM. Table 7 shows the effects of these sub-components. The model with the non-salient prototype had a gain of 1.2% in terms

Table 7
Ablation studies on internal components of class-wise non-salient features. NSP, NP, and CAM are the non-salient prototype, naive prototype, and class activation map, respectively.

Method	NP	CAM	NSP	$\rightarrow V_s$	$\rightarrow CV_s$	$\rightarrow S_s$	Avg
Baseline				28.2	60.8	40.0	43.0
NP	✓			33.1	58.4	44.8	45.4
CAM		✓		32.7	60.4	43.7	45.6
NSP			✓	33.8	57.8	48.2	46.6
NP + CAM	✓	✓		32.8	**61.2**	45.0	46.3
NSP + CAM		✓	✓	**34.1**	60.0	**49.5**	**47.9**

Table 8
Ablation studies on internal component of inter-frame non-salient centroid alignment. NSCA and NCA represent the non-salient centroid and naive centroid alignments, respectively.

Method	NSCA	NCA	$\rightarrow V_s$	$\rightarrow CV_s$	$\rightarrow S_s$	Avg
Baseline			28.2	**60.8**	40.0	43.0
NCA	✓		**32.8**	56.9	45.4	45.0
NSCA		✓	32.6	57.4	**50.0**	**46.7**

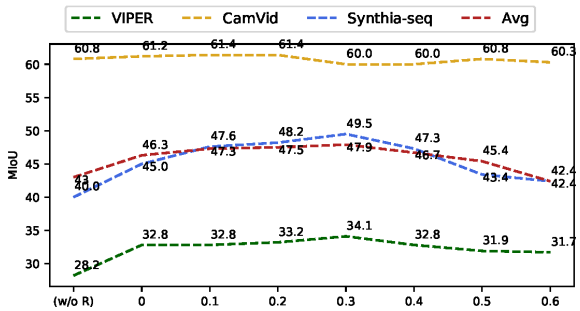

Figure 10: The hyperparameter evaluation of deciding the non-salient region. (w/o R) denotes the results without NSFR.

of the average performance compared to the model with the naive prototype. Meanwhile, the NSP + CAM model achieved an average mIoU of 47.9% and outperformed the NP + CAM model by 1.3% on average, indicating that the features in the non-salient region carry more generalized information. Second, as shown in Table 8, the proposed NSCA achieved an improvement of 1.7% in the average mIoU over the naive centroid alignment. These experiments demonstrate the effectiveness of the proposed components and non-salient region.

4.4.3. Non-salient region hyperparameter evaluation

The hyperparameter for determining the non-salient region is also important, where the hyperparameter α represents the filter percentage. For example, 30% of pixels in the feature map will be filtered when the hyperparameter is 0.3. As shown in Figure 10, the average performance was the best when the hyperparameter α was 0.3. Meanwhile, the performance reasonably increased and then decreased as α increased. First, the performance without NSFR was the worst. Then, the model with feature reasoning but without the non-salient region filter ($\alpha = 0$) improved the generalized performance, which verifies the effectiveness of the feature reasoning. Next, as the features of most salient regions that provided less generalizable information were filtered out, the performance increased when α increased. Finally, features carrying domain-invariant information were filtered, which led to decreased performance when α increased further.

Table 9
Time complexity comparison. ↓ represents that lower value is better and ↑ shows higher value is better.

Model	Runtime (s) ↓	GFLOPS ↓	Parameter (M) ↓	FPS ↑
CFFM[82]	0.0399	28.66	15.3	25.1
TMA[7]	0.0304	242.71	27.3	33.9
IBN[16]	0.0243	74.68	40.4	41.2
SW [62]	0.0317	74.63	40.4	31.5
ISW[17]	0.0280	74.65	40.4	35.7
Pinmem[13]	0.0275	78.37	40.5	36.4
Baseline	0.0310	149.31	40.4	32.3
NSCA	0.0318	168.68	45.7	31.4
NSFR	0.0331	190.43	46.3	30.2
NSFRC	0.0355	190.43	46.3	28.2

4.4.4. Time complexity analysis

We also present a time complexity comparison in the condition of the ResNet-50 backbone, including state-of-the-art methods and the proposed sub-modules. As shown in Table 9, CFFM had the lowest GFLOPS and parameters owing to the new backbone Transformer, while suffering from the lowest FPS owing to the complex calculation. Compared to the other VSS method TMA, our final method had lower GFLOPS and obtained a 12.7% improvement in the average mIoU with no significant difference in the FPS. Compared with the IGSS methods, there was no significant variance in the parameters and FPS, whereas our method achieved an average mIoU of at least 3.1%, thereby highlighting the efficacy of our method.

5. Limitations

Although our method achieved state-of-the-art performance, we reckon that some limitations need to be overcome. Our core idea employs a class-wise non-salient feature as prior information to select and enhance the feature channel adaptively, where the class-wise non-salient feature is constructed using a class centroid and CAM. As noted in RSSP [95], using one class prototype to present a class is not sufficient, as the classes can also be divided into different parts or originate from different domains. Thus, our work may be improved by fine-grained class prototype extraction; that is, by employing multiple prototypes for each category. Meanwhile, as a pioneering VGSS method, our approach mainly focuses on improving the generalized performance using video information, without considering an increase in consumption. Although the cost of our model is not significantly different from that of other state-of-the-art methods, it is a valuable aspect for improvement. Moreover, although

desirable, we could not perform our experiment on larger-scale datasets such as the AIO Drive dataset [96] owing to resource limitations.

6. Conclusions

We have introduced a new task to deal with dynamic scenes in real-world environments, namely: VGSS, which considers both continuous data and model generalizability. To the best of our knowledge, this task has not been previously studied. For the VGSS task, we proposed a novel method known as NSFRC. Specifically, we first defined the class-wise non-salient feature, which describes the features of the class-wise non-salient region that carry more generalizable information. We then proposed a class-wise NSFR strategy to select and enhance the generalizable channels adaptatively. Finally, we presented the inter-frame NSCA loss to alleviate temporally inconsistent and negative transfer problems in the VGSS task. Furthermore, we extended our method to the IGSS task. Extensive results on both the VGSS and IGSS tasks demonstrate the superiority of our NSFRC framework.

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] F. Lv, G. Lin, P. Liu, G. Yang, S. J. Pan, L. Duan, Weakly-supervised cross-domain road scene segmentation via multi-level curriculum adaptation, IEEE Trans. Circuit Syst. Vid. Tech. 31 (2020) 3493 – 3503.

[2] W. Shi, J. Xu, D. Zhu, G. Zhang, X. Wang, J. Li, X. Zhang, Rgb-d semantic segmentation and label-oriented voxelgrid fusion for accurate 3d semantic mapping, IEEE Trans. Circuit Syst. Vid. Tech. 32 (1) (2021) 183–197.

[3] Z. Jiang, Y. He, S. Ye, P. Shao, X. Zhu, Y. Xu, Y. Chen, J.-L. Coatrieux, S. Li, G. Yang, O2m-uda: Unsupervised dynamic domain adaptation for one-to-multiple medical image segmentation, Knowledge-Based Systems 265 (2023) 110378.

[4] L. Lu, Y. Xiao, X. Chang, X. Wang, P. Ren, Z. Ren, Deformable attention-oriented feature pyramid network for semantic segmentation, Knowledge-Based Systems 254 (2022) 109623.

[5] M. Liao, S. Tian, Y. Zhang, G. Hua, W. Zou, X. Li, Domain-invariant information aggregation for domain generalization semantic segmentation, Neurocomputing 546 (2023) 126273.

[6] J. Li, W. Wang, J. Chen, L. Niu, J. Si, C. Qian, L. Zhang, Video semantic segmentation via sparse temporal transformer, in: Proceedings of the 29th ACM International Conference on Multimedia, 2021, pp. 59–68.

[7] H. Wang, W. Wang, J. Liu, Temporal memory attention for video semantic segmentation, in: IEEE Int. Conf. Image Proc. (ICIP), IEEE, 2021, pp. 2254–2258.

[8] Y. Zhang, M. Ye, Y. Gan, W. Zhang, Knowledge based domain adaptation for semantic segmentation, Knowledge-Based Systems 193 (2020) 105444.

[9] Y. Yang, Q. Chen, Q. Liu, A dual-channel network for cross-domain one-shot semantic segmentation via adversarial learning, Knowledge-Based Systems (2023) 110698.

[10] M. Wang, S. Wang, Y. Wang, W. Wang, T. Liang, J. Chen, Z. Luo, Boosting unsupervised domain adaptation: A fourier approach, Knowledge-Based Systems 264 (2023) 110325.

[11] C. Yang, J. Xiao, Y. Ju, G. Qiu, K.-M. Lam, Improving robustness of single image super-resolution models with monte carlo method, in: IEEE Int. Conf. Image Proc. (ICIP), IEEE, 2023, pp. 2135–2139.

[12] D. Peng, Y. Lei, M. Hayat, Y. Guo, W. Li, Semantic-aware domain generalized segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2022, pp. 2594–2605.

[13] J. Kim, J. Lee, J. Park, D. Min, K. Sohn, Pin the memory: Learning to generalize semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2022, pp. 4350–4360.

[14] S. R. Richter, V. Vineet, S. Roth, V. Koltun, Playing for data: Ground truth from computer games, in: Proc. Eur. Conf. Comput. Vis. (ECCV), Amsterdam, The Netherlands, 2016, pp. 102–118.

[15] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, Nevada, USA, 2016, pp. 3213–3223.

[16] X. Pan, P. Luo, J. Shi, X. Tang, Two at once: Enhancing learning and generalization capacities via ibn-net, in: Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 464–479.

[17] S. Choi, S. Jung, H. Yun, J. T. Kim, S. Kim, J. Choo, Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2021, pp. 11580–11590.

[18] B. Shi, D. Zhang, Q. Dai, Z. Zhu, Y. Mu, J. Wang, Informative dropout for robust representation learning: A shape-bias perspective, in: International Conference on Machine Learning, PMLR, 2020, pp. 8828–8839.

[19] Z. Huang, H. Wang, E. P. Xing, D. Huang, Self-challenging improves cross-domain generalization, in: Proc. Eur. Conf. Comput. Vis. (ECCV), Springer, 2020, pp. 124–140.

[20] Y. Liu, J. Deng, X. Gao, W. Li, L. Duan, Bapa-net: Boundary adaptation and prototype alignment for cross-domain semantic segmentation, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2021, pp. 8801–8811.

[21] Y. Lu, Y. Luo, L. Zhang, Z. Li, Y. Yang, J. Xiao, Bidirectional self-training with multiple anisotropic prototypes for domain adaptive semantic segmentation, in: Proc. ACM Int. Conf. Multi. (ACMMM), 2022, pp. 1405–1415.

[22] J. Xiong, L.-M. Po, W. Y. Yu, Y. Zhao, K.-W. Cheung, Distortion map-guided feature rectification for efficient video semantic segmentation, IEEE Trans. Multim.

[23] D. Guan, J. Huang, A. Xiao, S. Lu, Domain adaptive video segmentation via temporal consistency regularization, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2021, pp. 8053–8064.

[24] Y. Xing, D. Guan, J. Huang, S. Lu, Domain adaptive video segmentation via temporal pseudo supervision, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Springer, 2022, pp. 621–639.

[25] Z. Fan, G. Hu, X. Sun, G. Wang, J. Dong, C. Su, Self-attention neural architecture search for semantic image segmentation, Knowledge-Based Systems 239 (2022) 107968.

[26] G. Hua, M. Liao, S. Tian, Y. Zhang, W. Zou, Multiple relational learning network for joint referring expression comprehension and segmentation, IEEE Trans. Multim.

[27] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern

Recognit. (CVPR), 2015, pp. 3431–3440.
[28] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Int. Conf. Medi. Img. Comp. Comp. Ass. Inter. (MICCAI), 2015, pp. 234–241.
[29] K. Sun, B. Xiao, D. Liu, J. Wang, Deep high-resolution representation learning for human pose estimation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2019, pp. 5693–5703.
[30] 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: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2021, pp. 6881–6890.
[31] R. Strudel, R. Garcia, I. Laptev, C. Schmid, Segmenter: Transformer for semantic segmentation, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2021, pp. 7262–7272.
[32] X. Ding, H. Chen, X. Zhang, J. Han, G. Ding, Repmlpnet: Hierarchical vision mlp with re-parameterized locality, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2022, pp. 578–587.
[33] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2017) 834–848.
[34] L.-C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, arXiv preprint arXiv:1706.05587.
URL https://arxiv.org/abs/1706.05587
[35] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 2881–2890.
[36] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, W. Liu, Ccnet: Criss-cross attention for semantic segmentation, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2019, pp. 603–612.
[37] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 7794–7803.
[38] M.-H. Guo, C.-Z. Lu, Q. Hou, Z.-N. Liu, M.-M. Cheng, S.-m. Hu, Segnext: Rethinking convolutional attention design for semantic segmentation, in: Advances in Neural Information Processing Systems.
[39] X. Zhu, Y. Xiong, J. Dai, L. Yuan, Y. Wei, Deep feature flow for video recognition, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 2349–2358.
[40] Y.-S. Xu, T.-J. Fu, H.-K. Yang, C.-Y. Lee, Dynamic video segmentation network, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 6556–6565.
[41] Y. Li, J. Shi, D. Lin, Low-latency video semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 5997–6005.
[42] J. Zhuang, Z. Wang, B. Wang, Video semantic segmentation with distortion-aware feature correction, IEEE Trans. Circuit Syst. Vid. Tech. 31 (8) (2020) 3128–3139.
[43] M. Ding, Z. Wang, B. Zhou, J. Shi, Z. Lu, P. Luo, Every frame counts: Joint learning of video segmentation and optical flow, in: Proc. AAAI Conf. Artif. Intell., Vol. 34, 2020, pp. 10713–10720.
[44] W. Liu, G. Lin, T. Zhang, Z. Liu, Guided co-segmentation network for fast video object segmentation, IEEE Trans. Circuit Syst. Vid. Tech. 31 (4) (2020) 1607–1617.
[45] S. Varghese, S. Gujamagadi, M. Klingner, N. Kapoor, A. Bar, J. D. Schneider, K. Maag, P. Schlicht, F. Huger, T. Fingscheidt, An unsupervised temporal consistency (tc) loss to improve the performance of semantic segmentation networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2021, pp. 12–20.
[46] T. Zhang, Z. Gao, Z. Liu, S. F. Hussain, M. Waqas, Z. Halim, Y. Li, Infrared ship target segmentation based on adversarial domain adaptation, Knowledge-Based Systems 265 (2023) 110344.
[47] Y. Yang, S. Soatto, Fda: Fourier domain adaptation for semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Seattle, WA, USA, 2020, pp. 4085–4095.
[48] W. Tranheden, V. Olsson, J. Pinto, 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.
[49] Q. ZHANG, J. Zhang, W. Liu, D. Tao, Category anchor-guided unsupervised domain adaptation for semantic segmentation, Advances in Neural Information Processing Systems 32 (2019) 435–445.
[50] Z. Wang, M. Yu, Y. Wei, R. Feris, J. Xiong, W.-m. Hwu, T. S. Huang, H. Shi, Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Seattle, WA, USA, 2020, pp. 12635–12644.
[51] M. Liao, G. Hua, S. Tian, Y. Zhang, W. Zou, X. Li, Exploring more concentrated and consistent activation regions for cross-domain semantic segmentation, Neurocomputing.
[52] W. Zou, R. Long, Y. Zhang, M. Liao, Z. Zhou, S. Tian, Dual geometric perception for cross-domain road segmentation, Displays (2022) 102332.
[53] Y. Zhang, S. Tian, M. Liao, W. Zou, C. Xu, A hybrid domain learning framework for unsupervised semantic segmentation, Neurocomputing 516 (2023) 133–145.
[54] Y. Li, L. Yuan, N. Vasconcelos, Bidirectional learning for domain adaptation of semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Long Beach, CA, USA, 2019, pp. 6936–6945.
[55] Q. Zhou, Z. Feng, Q. Gu, J. Pang, G. Cheng, X. Lu, J. Shi, L. Ma, Context-aware mixup for domain adaptive semantic segmentation, IEEE Trans. Circuit Syst. Vid. Tech.
[56] P. Zhang, B. Zhang, T. Zhang, D. Chen, Y. Wang, F. Wen, Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2021, pp. 12414–12424.
[57] Y. Wang, L. Qi, Y. Shi, Y. Gao, Feature-based style randomization for domain generalization, IEEE Trans. Circuit Syst. Vid. Tech. 32 (8) (2022) 5495–5509.
[58] X. Yue, Y. Zhang, S. Zhao, A. Sangiovanni-Vincentelli, K. Keutzer, B. Gong, Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2019, pp. 2100–2110.
[59] J.-Y. Zhu, T. Park, P. Isola, A. A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2017, pp. 2223–2232.
[60] J. Huang, D. Guan, A. Xiao, S. Lu, Fsdr: Frequency space domain randomization for domain generalization, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2021, pp. 6891–6902.
[61] D. Peng, Y. Lei, L. Liu, P. Zhang, J. Liu, Global and local texture randomization for synthetic-to-real semantic segmentation, IEEE Trans. Image Process. 30 (2021) 6594–6608.
[62] X. Pan, X. Zhan, J. Shi, X. Tang, P. Luo, Switchable whitening for deep representation learning, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2019, pp. 1863–1871.
[63] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 2921–2929.
[64] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2017, pp. 618–626.
[65] A. Chattopadhay, A. Sarkar, P. Howlader, V. N. Balasubramanian, Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks, in: IEEE Win. Conf. App. Compu. Vis. (WACV), IEEE, 2018, pp. 839–847.
[66] Y. Wang, J. Zhang, M. Kan, S. Shan, X. Chen, Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2020, pp. 12275–12284.
[67] L. Xu, W. Ouyang, M. Bennamoun, F. Boussaid, F. Sohel, D. Xu, Leveraging auxiliary tasks with affinity learning for weakly supervised semantic segmentation, in: Proc. IEEE/CVF Int. Conf. Comput.

Vis. (ICCV), 2021, pp. 6984–6993.
[68] D. Du, J. Chen, Y. Li, K. Ma, G. Wu, Y. Zheng, L. Wang, Cross-domain gated learning for domain generalization, International Journal of Computer Vision 130 (11) (2022) 2842–2857.
[69] Z. Jiang, Y. Li, C. Yang, P. Gao, Y. Wang, Y. Tai, C. Wang, Prototypical contrast adaptation for domain adaptive semantic segmentation, in: Proc. Eur. Conf. Comput. Vis. (ECCV), Springer, 2022, pp. 36–54.
[70] L. Zhang, X. Li, A. Arnab, K. Yang, Y. Tong, P. Torr, Dual graph convolutional network for semantic segmentation, in: Proc. Brit. Mach. Vis. Conf. (BMVC), British Machine Vision Association, 2019.
[71] Y. Lu, Y. Chen, D. Zhao, B. Liu, Z. Lai, J. Chen, Cnn-g: Convolutional neural network combined with graph for image segmentation with theoretical analysis, IEEE Trans. Cogn. Dev. Sys. 13 (3) (2020) 631–644.
[72] C. Li, D. Du, L. Zhang, L. Wen, T. Luo, Y. Wu, P. Zhu, Spatial attention pyramid network for unsupervised domain adaptation, in: Proc. Eur. Conf. Comput. Vis. (ECCV), Glasgow, UK, 2020, pp. 481–497.
[73] S. Wan, C. Gong, P. Zhong, B. Du, L. Zhang, J. Yang, Multiscale dynamic graph convolutional network for hyperspectral image classification, IEEE Trans. Geo. Rem. Sens. 58 (5) (2019) 3162–3177.
[74] J. Ye, J. He, X. Peng, W. Wu, Y. Qiao, Attention-driven dynamic graph convolutional network for multi-label image recognition, in: Proc. Eur. Conf. Comput. Vis. (ECCV), Springer, 2020, pp. 649–665.
[75] H. Hu, D. Ji, W. Gan, S. Bai, W. Wu, J. Yan, Class-wise dynamic graph convolution for semantic segmentation, in: Proc. Eur. Conf. Comput. Vis. (ECCV), Springer, 2020, pp. 1–17.
[76] P. Ghosh, Y. Yao, L. Davis, A. Divakaran, Stacked spatio-temporal graph convolutional networks for action segmentation, in: IEEE Win. Conf. App. Compu. Vis. (WACV), 2020, pp. 576–585.
[77] S. Park, N. Kwak, Analysis on the dropout effect in convolutional neural networks, in: Asian conference on computer vision, Springer, 2016, pp. 189–204.
[78] Y. Chen, M. Rohrbach, Z. Yan, Y. Shuicheng, J. Feng, Y. Kalantidis, Graph-based global reasoning networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2019, pp. 433–442.
[79] X. Li, X. Li, A. You, L. Zhang, G. Cheng, K. Yang, Y. Tong, Z. Lin, Towards efficient scene understanding via squeeze reasoning, IEEE Trans. Image Process. 30 (2021) 7050–7063.
[80] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, T. Brox, Flownet 2.0: Evolution of optical flow estimation with deep networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 2462–2470.
[81] Z. Wang, Z. Dai, B. Póczos, J. Carbonell, Characterizing and avoiding negative transfer, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2019, pp. 11293–11302.
[82] G. Sun, Y. Liu, H. Ding, T. Probst, L. Van Gool, Coarse-to-fine feature mining for video semantic segmentation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2022, pp. 3126–3137.
[83] G. J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla, Segmentation and recognition using structure from motion point clouds, in: Proc. Eur. Conf. Comput. Vis. (ECCV), Springer, 2008, pp. 44–57.
[84] G. Neuhold, T. Ollmann, S. Rota Bulo, P. Kontschieder, The mapillary vistas dataset for semantic understanding of street scenes, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2017, pp. 4990–4999.
[85] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, T. Darrell, Bdd100k: A diverse driving dataset for heterogeneous multitask learning, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2020, pp. 2636–2645.
[86] S. R. Richter, Z. Hayder, V. Koltun, Playing for benchmarks, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2017, pp. 2213–2222.
[87] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, A. M. Lopez, The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, Nevada, USA, 2016, pp. 3234–3243.
[88] Y. Li, D. Ma, Z. An, Z. Wang, Y. Zhong, S. Chen, C. Feng, V2x-sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving, IEEE Robot. Autom. Letters 7 (4) (2022) 10914–10921.
[89] Y. Cabon, N. Murray, M. Humenberger, Virtual kitti 2, arXiv preprint arXiv:2001.10773.
URL https://arxiv.org/abs/2001.10773
[90] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 770–778.
[91] N. Ma, X. Zhang, H.-T. Zheng, J. Sun, Shufflenet v2: Practical guidelines for efficient cnn architecture design, in: Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 116–131.
[92] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 4510–4520.
[93] C. Sakaridis, D. Dai, L. Van Gool, Semantic foggy scene understanding with synthetic data, Int. Jour. Compu. Vis. 126 (2018) 973–992.
[94] G. Varma, A. Subramanian, A. Namboodiri, M. Chandraker, C. Jawahar, Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments, in: IEEE Win. Conf. App. Compu. Vis. (WACV), IEEE, 2019, pp. 1743–1751.
[95] T. Zhou, W. Wang, E. Konukoglu, L. Van Gool, Rethinking semantic segmentation: A prototype view, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2022, pp. 2582–2593.
[96] X. Weng, Y. Man, D. Cheng, J. Park, M. O'Toole, K. Kitani, All-In-One Drive: A Large-Scale Comprehensive Perception Dataset with High-Density Long-Range Point Clouds, arXiv.

Declaration of interests

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

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: