# Train One, Generalize to All: Generalizable Semantic Segmentation from Single-Scene to All Adverse Scenes

Ziyang Gong[*]
Sun Yat-sen University
Zhuhai, China
gongzy23@mail2.sysu.edu.cn

Fuhao Li[*]
Wuhan University of Science and Technology
Wuhan, China
lfh@wust.edu.cn

Yupeng Deng[*]
National University of Singapore
Singapore
ydeng@u.nus.edu

Wenjun Shen
Wuhan College
Wuhan, China
shenwj129@163.com

Xianzheng Ma[†]
Shanghai AI Laboratory
Shanghai, China
maxianzheng@pjlab.org.cn

Zhenming Ji[†]
Sun Yat-sen University
Zhuhai, China
jizhm3@mail.sysu.edu.cn

Nan Xia
South China University of Technology
Guangzhou, China
menanxia@mail.scut.edu.cn

## ABSTRACT

Unsupervised Domain Adaptation (UDA) for semantic segmentation has received widespread attention for its ability to transfer knowledge from the source to target domains without a high demand for annotations. However, semantic segmentation under adverse conditions still poses significant challenges for autonomous driving, as bad weather observation data may introduce unforeseeable problems. Although previous UDA works are devoted to adverse scene tasks, their adaptation process is redundant. For instance, unlabeled snow scene training data is a must for the model to achieve fair segmentation performance in snowy scenarios. We propose calling this type of adaptation process the Single to Single (STS) strategy. Clearly, STS is time-consuming and may show weaknesses in some comprehensive scenes, such as a night scene of sleet. Motivated by the concept of Domain Generalization (DG), we propose the Single to All (STA) model. Unlike DG, which trains models on one or multiple source domains without target domains, the STA model is based on UDA and employs one source domain, one target domain, and one introduced domain to achieve generalization to all adverse conditions by training on a single-scene dataset. Specifically, the STA model is advantageous as it learns from the source domain, reserves the style factors via a Reservation domain, and adapts the unified factors by the Randomization module. An Output Space Refusion module is also further incorporated to strengthen STA. Our STA achieves state-of-the-art performance in the Foggy Driving benchmark and demonstrates great domain generalizability in all conditions of the ACDC and Foggy Zurich benchmarks.

## CCS CONCEPTS

• **Computing methodologies** → **Computer vision tasks**; • **Human-centered computing** → *Visualization application domains*; • **Applied computing** → *Transportation.*

## KEYWORDS

Semantic Segmentation, Unsupervised Domain Adaptation, Domain Generalization, Adverse scenes understanding

## 1 INTRODUCTION

Standing at the intersection of semantic segmentation and autonomous driving, we believe that unsupervised learning will dominate in semantic segmentation tasks in the future, given the time-consuming and laborious annotation work required for supervised training. Achieving semantic segmentation tasks under adverse scenarios such as fog, night, rain, and snow is crucial for autonomous driving. However, most existing unsupervised domain adaptation (UDA) methods are only suitable for controlled environments and are vulnerable[22, 40] to domain distribution shift.

To address this issue, excellent works have been proposed to tackle the challenge of understanding adverse weather.[3, 24, 30]. Although a few methods demonstrate generalization in adverse scenes, their models are trained via the redundant Single to Single

[*] Equal contributions. The listing order is random.
[†] Corresponding authors.

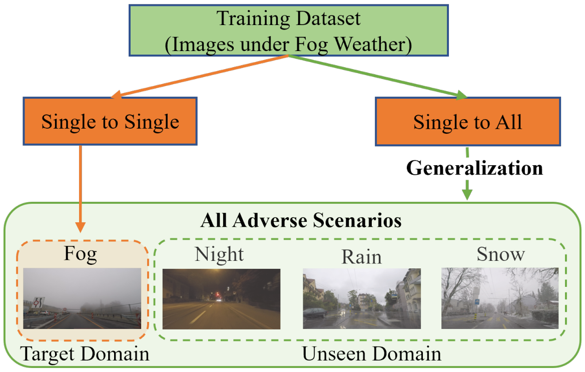

**Figure 1: Difference between STS and STA.**

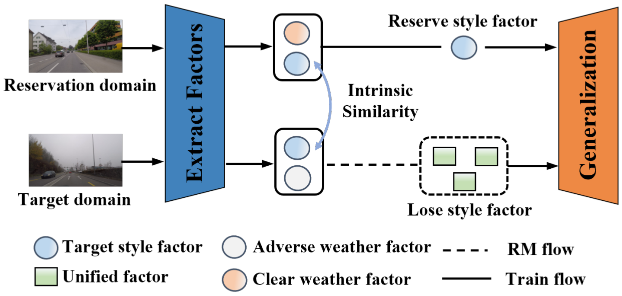

**Figure 2: The factor-level process of STA. RD reserves style factors and RM creates unified factors. The model needs to learn both unified factors and style factors to achieve Single to All.**

(STS) generalization process. We argue that STS is time-consuming and may not be effective in dealing with more complex and comprehensive scenarios, such as a rainy or snowy night.

To overcome this limitation, we propose to achieve Domain Generalization (DG) in UDA adverse scene tasks. Specifically, we aim to build a model trained on a single scene dataset that can generalize to all adverse scenarios. The key difference between our approach and STS is that STS generalizes from the source domain to the target domain, whereas our aim is to generalize from the source domain to the target domain and unseen domains, as shown in Fig 1.

Our motivation for the Single to All (STA) model is based on pioneering methods that have demonstrated two gaps, style and weather gaps, between the source and target domains in Unsupervised Domain Adaptation (UDA) for semantic segmentation tasks. These two domain images are often captured under different cities and weather conditions, leading to weather differences in weather, illuminance, and contrast, and style differences in roads, sidewalks, and other classes.

Based on this, we reasonably disentangle the features of images to style-related features called style factors and weather-related features called weather factors. Similarly, we set weather factors to represent the weather of images and style factors to represent the content information of images.

Following this line of thought, we can view domain adaptation as a process in which a model learns from the factors of the source domain and then adapts to the factors of the target domains. To achieve the Single to All (STA) approach on ACDC benchmarks, we propose to achieve mutual similarity of the style and weather factors of all adverse scenes. With this highly mutual similarity, the model can learn a set of factors that are common to all adverse scenes from one scene, which allows it to adapt to new scenes more easily and achieve the STA approach.

To this end, we propose the Randomization module (RM) employing style transferring to change all the factors of an image to other unified factors. Intuitively, we argue that transformed images have highly mutual similarity because they are processed by the same transformation flow with the same parameters. The RM plays the role of a bridge to connect the factors of all different scenes into unified factors. If our model can learn these unified factors from an arbitrary adverse scene and achieve good results on the corresponding scene benchmark (STS), it can also achieve cross-scene generalization to all adverse scenes from one scene. However, we are concerned that the strong transformation of the RM may have some side effects. Since the RM mainly focuses on transforming weather factors, it may cause a significant loss of style factors. To achieve further unification of the factors, we propose to employ other methods to improve the similarity of the style factors separately.

We note that in our experiments, all target domain images are captured in the same city of Zurich, but with different adverse scenes. This means that although the scenes of these images are different, the style factors are intrinsically similar to each other due to the similar architectural style of Zurich, such as the roads and sidewalks mentioned above. Therefore, as shown in Fig 2, we need to reserve the style factors to ensure intrinsic similarity, while altering all target factors to unified factors to create a high similarity so that we can achieve Single to All.

To preserve the style factors while mitigating the adverse effects of external influences, we propose the Reservation domain (RD). The RD is a strategy that inputs clear weather images (ACDC-ref) containing more clear style factors without the influence of adverse weather. This approach ensures that our model can learn enough mutually similar style factors and unified factors. To further improve the performance of our model, we propose an alternative training strategy that runs RD and RM alternatively. By doing so, our model can learn style factors from the clear images and unified factors from the transformed images, leading to better generalization performance. We also fusion a multi-layer Output Space Refusion module in our framework to reinforce the model to learn the knowledge of these classes in detail.

Based on our discussion above, there are three key assumptions to achieve STA: (1) RM is essential, as it can significantly improve the mutual similarity of factors to unify different scenes. (2) RD, which can preserve the style factors, is also essential, as the Randomization module may lose the style factors due to the strong style transform. (3) Based on the unification by RM and RD, the STA model can perform well on every benchmark by STS.

In the subsequent Experiments section, we will conduct explicit experiments on ACDC benchmarks (fog, night, snow, and rain), Foggy Zurich, Foggy Driving, and Night Zurich to show the generalizability of STA and validate our assumptions. The main contributions of our works are as follows: (1) To the best of our knowledge, we are the first to achieve domain generalization in UDA for adverse semantic segmentation tasks. (2) We develop a novel STA framework that can adapt to all adverse conditions by training on a single-scene dataset. (3) Our STA model achieves state-of-the-art performance in the Foggy Driving benchmark and demonstrates superior performance than our baseline in all adverse conditions of ACDC and Foggy Zurich benchmarks.

## 2 RELATED WORK

**Domain Generalization (DG)** DG methods can be classified into multi-source and single-source. Multi-source methods use multiple source domains for training and evaluating performance on unseen target domains, utilizing techniques such as domain invariant feature learning [10], meta-learning [41], and invariant risk minimization [2]. Single-source methods [13], on the other hand, only use one domain for training and rely mainly on adversarial data augmentation [33]. Since all domain generalization tasks [34] aim to adapt to unseen domains, we aim to incorporate this principle into UDA tasks by using a single source domain and a target domain to adapt to other unseen adverse domains.

**Stlye Transfer** Style features have been widely explored in style transfer [9, 19, 56] to transform the image style to improve the performance of models. Both UDA and DG approaches leverage the style transfer to increase model robustness by generating a diversity of data and highlighting the contour of images through methods such as texture underfitting [54], swapping [44], and mixing [57]. In STA, we use a style embedding method [21] to obtain random factors. Compared to other methods, this method can significantly reduce the transferring time obviously through random and simulated sampling.

**Adverse Scenarios UDA** To minimize the domain discrepancy, Domain Adaptation (DA) has become a significant research area. However, due to the scarcity of pixel-level annotations, UDA has become the primary method for narrowing the domain gaps. To address the challenges generated by domain shift, many UDA methods have been proposed that focus on adversarial training [15, 29, 48, 58] or self-training [16, 18, 46]. These methods aim to reduce the domain gap by minimizing the statistical distance between domains by using techniques such as maximum mean discrepancy, correlation alignment, or entropy minimization. In adversarial training, a domain discriminator is trained in a generative adversarial network (GAN) [14] framework to promote domain-invariant features or outputs. Self-training approaches generate pseudo-labels [23] for the target domain based on predictions obtained using confidence thresholds or pseudo-label prototypes [31]. Other UDA strategies include using pretext tasks [5, 50], following adaptation curriculums [7, 37], and leveraging the domain-robustness of Transformers [16, 17, 43]. These UDA methods have shown promising results in various computer vision tasks, despite the scarcity of pixel-level annotations.

Adaptation to adverse scenarios, such as fog, night, rain, and snow, is highly relevant for the robust perception in autonomous driving. However, there are significantly fewer works that are capable of adapting to adverse domains [3, 11, 20, 24, 25] compared to normal UDA tasks. Several methods have been proposed to improve performance on real-world adverse weather tasks, such as generating synthetic data [38, 47], using adversarial style transfer as pre-processing steps before predicting with source-domain models [36, 42], and leveraging shared characteristics between different domains [11, 24]. However, most of the current UDA tasks focus on specialized training for adverse scenes using STS. Our approach STA, achieves domain generalization in UDA tasks, by training one scene and generalizing to all conditions.

**Algorithm 1** STA algorithm

```
Require: Samples $D_S, D_R, D_T$, initialized network $f_\theta$, two modules, $F_t$
    and $F_r$, and FDs weights and loss, $\alpha_{FDs}$ and $L_{FDs}$
for i = 0 to $N$ do
    update/initialize teacher network $f_\phi$
    $X_S, Y_S \sim D_S$
    $X_R \sim D_R, X_T \sim D_T$
    if $i\%2 == 0$ then
        $\widehat{Y}_R \leftarrow f_\theta\left(X_R\right)$
        $\widehat{Y}'_R \leftarrow F_r\left(\widehat{Y}_R, X_R\right)$                  // Refusion $F_r$
        $X_S^R, Y_S^R \leftarrow$ Both images, pseudo-labels and weights from mixing
    $X_S, Y_S, X_R$ and $\widehat{Y}'_R$                                   // Mix $R$ with $S$
        $\widehat{Y}_S \leftarrow f_\theta\left(X_S\right), \widehat{Y}_S^R \leftarrow f_\phi\left(X_S^R\right)$// Compute predictions
        $l \leftarrow L\left(\widehat{Y}_S, Y_S, \widehat{Y}_S^R, Y_S^R\right) + \alpha_{FDs} L_{FDs}$          // Compute loss
    else
        $X_{T'} \leftarrow F_t\left(X_T\right)$                          // Randomization $F_t$
        $\widehat{Y}_{T'} \leftarrow f_\theta\left(X_{T'}\right)$
        $\widehat{Y}'_T \leftarrow F_r\left(\widehat{Y}_{T'}, X_{T'}\right)$                   // Refusion $F_r$
        $X_S^{T'}, Y_S^{T'} \leftarrow$ Both images, pseudo-labels and weights from mixing
    $X_S, Y_S, X_{T'}$ and $\widehat{Y}'_T$.                     // Mix translated $T$ with $S$
        $\widehat{Y_S} \leftarrow f_\theta\left(X_S\right), \widehat{Y}_S^{T'} \leftarrow f_\phi\left(X_S^{T'}\right)$        // Compute predictions
        $l \leftarrow L\left(\widehat{Y}_S, Y_S, \widehat{Y}_S^{T'}, Y_S^{T'}\right) + \alpha_{FDs} L_{FDs}$         // Compute loss
    end if
    Compute $\nabla_\theta l$ by backpropagation and apply SGD on $\theta$
end for
```

## 3 METHOD

### 3.1 Overview of the Proposed Framework

Given the images from the source domain $X_S = \left\{x_S^{(i)}\right\}_{i=1}^{N_S} \in \mathrm{R}^{\mathrm{H}\times\mathrm{W}\times 3}$ and ground-truth labels $Y_S = \left\{y_S^{(i)}\right\}_{i=1}^{N_S} \in \mathrm{R}^{\mathrm{H}\times\mathrm{W}\times\mathrm{C}}$, where S is the source domain, $N_S$ is the number of the $x_S$ and $y_S$, $H$ and $W$ are the height and width of the images, and the $C$ is the number of categories. The images without labels $X_T = \left\{x_T^{(i)}\right\}_{i=1}^{N_T} \in \mathrm{R}^{\mathrm{H}\times\mathrm{W}\times 3}$ from target domain $T$. We introduce a Reservation domain $R$ as an intermediate domain with unlabeled images $X_R = \left\{x_R^{(i)}\right\}_{i=1}^{N_R} \in \mathrm{R}^{\mathrm{H}\times\mathrm{W}\times 3}$. Besides, we leverage the Randomization module $F_t$ to transform the $T$ to a transformed domain $T'$. We fusion $S$, $T$, $T'$, $R$, $F_t$ and an Output Space Refusion module $F_r$ to build our STA framework. The whole model architecture is shown in Fig 3.

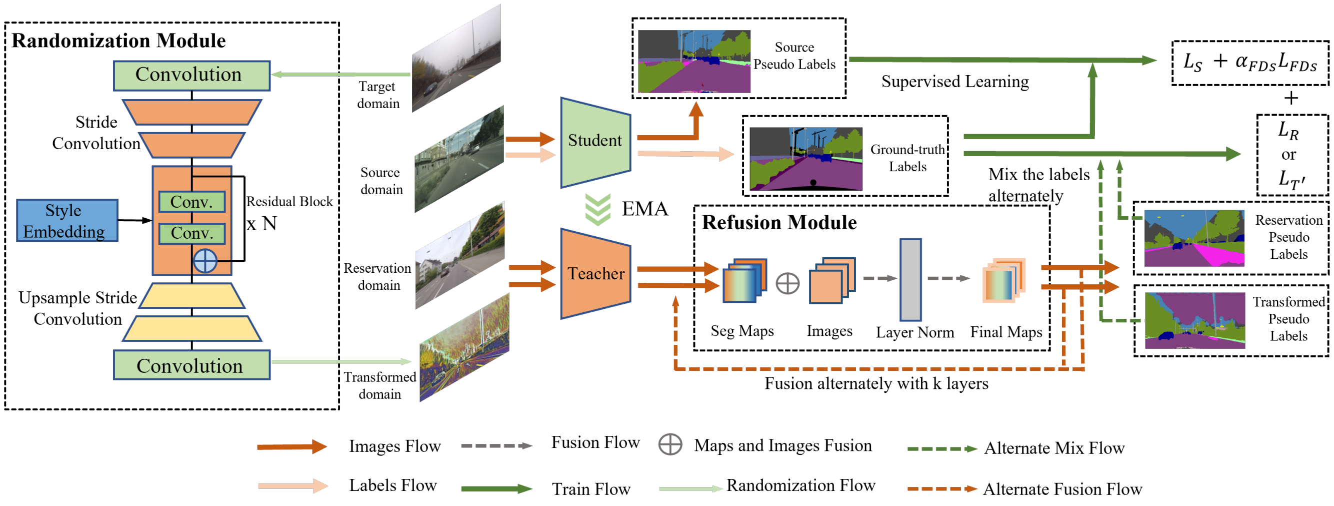

**Figure 3: The target domain is transformed into another domain by RM. Then, source domain images enter the student network, and the Reservation and transformed domain images enter the teacher network. The Refusion module will help the teacher network produce more robust pseudo labels. The pseudo labels of the Reservation and transformed domains will alternately mix with the ground-truth labels of the source domain images to calculate two cross-entropy losses, $L_R$ and $L_T'$, while the student network also calculates the cross-entropy loss, $L_S$. For the final loss, we follow the Feature Distance [16] loss, $L_{FDs}$, and sum all the above losses to train STA.**

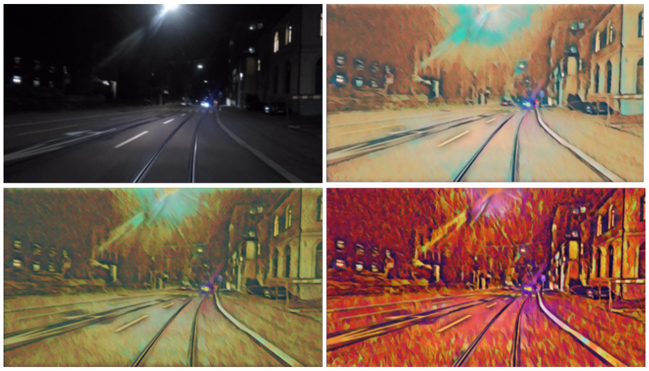

**Figure 4: Example images from original and transformed ACDC-night dataset. The original image is on the top left and the image with $\eta$=0.5 is on the top right. The second-row images, from left to right, are transformed with $\eta$=0.3 and $\eta$=0.8 respectively.**

## 3.2 Randomization by Style Augmentation

We employ the strategy of Style Augmentation (SA) [21] to construct our Randomization module for STA. To better understand our STA, let us first recap SA. SA is a method for transforming an image $x_T$ to another random style image $x_{T'}$. As discussed earlier, one of the drawbacks of STS is time-consuming training, whereas the most prominent advantage of SA over other style transfer methods is its low computational cost which suits our requirements. SA is based on the style transfer network of Ghiasi et al. [12] which is trained on the PBN dataset [1] and leverages a style prediction network to generate a style embedding $s$. SA replaces the prediction network with a new method of directly sampling $s$ from a multivariate normal distribution, which has the same mean and covariance as the PBN dataset. This way, only a portion of the PBN is required resulting in much lower computational expenses. To obtain the final output feature maps, the style embedding and input images are passed through the style transfer network $P$, which is shown in the Randomization module of Fig 3.

However, simply incorporating $s$ and $x_T$ does not always yield the best performance, as it can not flexibly adjust some image attributes, such as brightness and contrast. For instance, high-brightness augmentation is more useful than darkening augmentation in a night scene. Thus, it is necessary to introduce a parameter $\eta$ to constrain the strength of SA by linearly interpolating the style embedding of images $s_T$ with randomly sampling style embedding. The final output embedding $z$ is the interpolation sum:

$$z = \eta N(\mu, \Sigma) + (1 - \eta)x_T \tag{1}$$

where $\mu$, $\Sigma$ are the mean vector and covariance matrix of $s$:

$$\mu = \mathbb{E}_s[s] \tag{2}$$

$$\Sigma_{i,j} = \mathrm{Cov}\left[s_i, s_j\right] \tag{3}$$

The Randomization module takes both the final $z$ and $x_T$ as inputs to generate feature maps. It is worth noting that except for the first Convolution block in Fig 3, every other block in this module employs conditional instance normalization [9]. This technique allows for the shifting and reshaping of activation channels based on style embedding. Therefore, the output feature maps $m$ from the Randomization module can be expressed as:

$$m = \frac{\gamma(P(x_T, z) - \mu)}{\delta} + \beta \tag{4}$$

The mean and standard deviation across the feature map spatial axes are represented by $\mu$ and $\delta$, respectively, while $\gamma$ and $\beta$ represent the weight and bias obtained from the style transformer network. In STA, we leverage SA to conduct factor randomization on target domains. To ensure a sensible effect under different adverse scenes, we set different values of $\eta$ to control factors which are shown in Fig 4. We also combine SA with our Output Space

[1] https://www.kaggle.com/c/painter-by-numbers

Refusion module and the Reservation domain to maximize the performance of the STA. More details about these modules will be discussed later.

## 3.3 Training Strategies for STA

The architecture of STA is illustrated in Figure 3. The student network, denoted as $f_\theta$, is the main training network, while the teacher network, denoted as $f_\Phi$, does not participate in the training process with gradient backpropagation. To achieve the goal of preserving the style factors of the target domain while transforming the target factors into unified factors, we use RM to obtain the unified factors by inputting adverse scene images and employ RD to obtain similar style factors by reserving clear weather images in the RD.

We first focus on the teacher network. The teacher network does not run simultaneously. In alternate training of STA, the RD works during even iterations, while the transformed domain runs at other times. We believe that it is crucial to reserve the style factors of the target domain, and alternate training can help STA learn style factors and unified factors separately, improving training efficiency and achieving revenue maximization by learning enough style factors without redundant disturbances.

**Output Space Refusion Module** However, the Randomization module may blur the texture and intensify the contour of images, leading to the disentanglement of the relationship between texture and content information. This can negatively impact the segmentation of some classes whose texture is highly related to the context [21], resulting in the loss of content information.

Therefore, since the final aim of the teacher network is to produce pseudo-labels to facilitate the training of student networks, we legitimately incorporate the Output Space Refusion module $F_r$ with a multi-layer mechanism to STA. This will strengthen the learning of those easily lost classes.

Several works [27, 35] have shown that feature reuse is an effective approach to improving model performance. However, these methods mainly focus on reuse features in high-dimensional space. In contrast, our Refusion module operates in the output space by combining the original images with the semantic segmentation maps in a low-dimensional space that contains information and context spatially and locally [48].

The multi-layer mechanism helps extract robust features and cumulatively strengthens the learning of some classes. However, overusing the Refusion module may harm performance, as some features are not useful, and certain classes may be over-learned. To regulate the strength of the Refusion module, we introduce a hyperparameter $\lambda$. We denote our kth layer of $F_r$, as $F_r^{(k)}$, the input of the first layer as $x_R^{(i)}$ and the input of the (k+1)th layer as the normalized summation $Sum_{LN}$ of the output of the kth layer and $x_R^{(i)}$. The process is shown as follows:

$$F_r^{(k+1)} = Sum_{LN}(\lambda_k f_\phi^{(k)}(x_{T'/R}^{(i)}), x_{T'/R}^{(i)}) \tag{5}$$

After the refusion model, the pseudo labels will be generated. Our strategy to obtain the pseudo labels follows the Self-Training(ST) [46, 55] approaches, which are effective and adaptable to the domain adaptation. We suppose that pseudo labels of the $T'$ or R are $\hat{Y}_{T'/R}^{(i,c)}$:

$$\hat{Y}_{T'/R}^{(i,c)} = \begin{cases} 1, & \text{if} \quad c = \underset{c'}{\text{argmax}} \, f_\phi\left(x_{T'/R}^{(i)}\right)^{(c')} \\ 0, & \end{cases} \tag{6}$$

where $f_\phi\left(x_{T'/R}^{(i)}\right)^{(c)}$ represents the softmax probability of pixel $x_{T'/R}^{(i)}$ belonging to the cth class. To improve the quality of the pseudo labels, we use the confidence estimate mechanism proposed in [45] to improve the quality of pseudo labels. Suppose to set a threshold $\tau$ and $q^{(i,c)}$ to measure the ratio of the pixels surpass the $\tau$:

$$q_{T'/R}^{(i,c)} = \frac{\sum_{H,W}(max_{c'} f_\phi(x_{T'/R}^{(h,w,c',i)}) > \tau)}{H \cdot W} \tag{7}$$

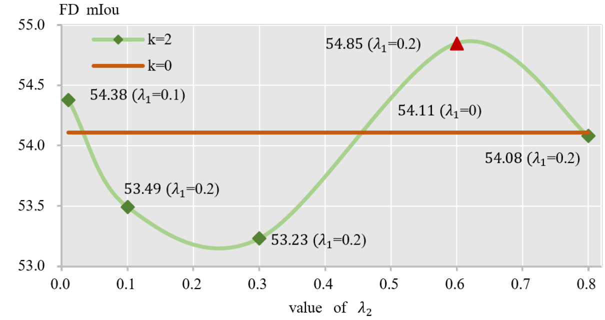

**Figure 5: Alation study on $\lambda$ of Refusion module on Foggy Driving-test dataset. The orange line is STA without Refusion and the abscissa axis is the second layer $\lambda_2$.**

We also employ ClassMix [32] in both two domains to mix with the source domain as a data augmentation to produce more robust pseudo labels. In STA, the images, labels, and weights will be mixed alternately for training. And we calculate the cross-entropy losses of these two domains are $L_{T'}$ and $L_R$.

$$L_{T'/R} = -\sum_{h\in H, w\in W} \sum_{c\in C} q_{T'/R}^{(i,c)} (Y_S^{T'/R})^{(h,w,c)} log\, f_\theta \left((X_S^{T'/R})^{(h,w,c)}\right) \tag{8}$$

where $(Y_S^{T'/R})^{(h,w,c)}$ and $(X_S^{T'/R})^{(h,w,c)}$ means mixed labels and images.

For the student network $f_\theta$, due to STA has the tendency of willing to learn the classes with high appearance frequency rather than the rarely appearing classes, we employ the Rare Class Sampling (RCS) [16] to conduct an upsampling for the rare class to ensure STA can learn more knowledge about them. To avoid the overfitting of STA, we follow the Feature Distance (FDs) [16] mechanisms which can strengthen the memories of previously learned knowledge to calculate the feature loss $L_{FDs}$ with a constrained parameter $\alpha_{FDs}$ which is set as 0.005. And we calculate the cross-entropy loss $L_s$ of the source domain.

$$L_S = -\sum_{h\in H, w\in W} \sum_{c\in C} Y_S^{(h,w,c)} log\, f_\theta \left(X_S^{(h,w,c)}\right) \tag{9}$$

Due to teacher networks not participating in training, the student network will share parameters with the two teachers. At the beginning of every new iteration of training, the student network will update the knowledge to the teacher networks by the exponential moving average weights (EMA) [45] which can improve the quality of pseudo-labels and mitigate confirmation bias between the source domain and the target domain. The whole algorithm flow is shown in Alg 1 Finally, the summary loss of the whole STA framework is:

$$L = L_S + L_{T'/R} + \alpha_{FDs} L_{FDs} \tag{10}$$

# 4 EXPERIMENTS

## 4.1 Datasets

**Cityscapes [6]** is a real-world dataset having 5,000 images (2975train, 500 val,1525test) of driving scenes in 50 different urban. It also has 19 categories of dense pixel annotations (97% coverage), 8 of which have instance-level segmentation.

**ACDC [40]** contains 4,006 images (400train, 100val (106 in night), 500test) distributed evenly across four common adverse weather conditions: fog, night, rain, and snow. Each unfavorable condition image has a high-quality

**Table 1: Quantitative results of mIou on all conditions benchmark of ACDC. We compare STA with other previous excellent works. All of the methods are trained by STS.**

| Method | Road | S.walk | Build | Wall | Fence | Pole | Tr.Light | Sign | Veget | Terrain | Sky | Person | Rider | Car | Truck | Bus | Train | M.bike | Bike | mIou |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Cityscapes to ACDC-all condition | | | | | | | | | | | | | | | | | | | | |
| AdaptSegNet[48] | 69.4 | 34 | 52.8 | 13.5 | 18 | 4.3 | 14.9 | 9.7 | 64 | 23.1 | 38.2 | 38.6 | 20.1 | 59.3 | 35.6 | 30.6 | 53.9 | 19.8 | 33.9 | 33.4 |
| BDL[28] | 56 | 32.5 | 68.1 | 20.1 | 17.4 | 15.8 | 30.2 | 28.7 | 59.9 | 25.3 | 37.7 | 28.7 | 25.5 | 70.2 | 39.6 | 40.5 | 52.7 | 29.2 | 38.4 | 37.7 |
| FDA[53] | 73.2 | 34.7 | 59 | 24.8 | 29.5 | 28.6 | 43.3 | 44.9 | 70.1 | 28.2 | 54.7 | 47 | 28.5 | 74.6 | 44.8 | 52.3 | 63.3 | 28.3 | 39.5 | 45.7 |
| DANNet(DeepLabV2)[51] | 82.9 | 53.1 | 75.3 | 32.1 | 28.2 | 26.5 | 39.4 | 40.3 | 70 | 39.7 | 83.5 | 42.8 | 28.9 | 68 | 32 | 31.6 | 47 | 21.5 | 36.7 | 46.3 |
| DANIA(DeepLabV2)[52] | 87.8 | 57.1 | 80.3 | 36.2 | 31.4 | 28.6 | 49.5 | 45.8 | 76.2 | 48.8 | 90.2 | 47.9 | 31.1 | 75.5 | 36.5 | 36.5 | 47.8 | 32.5 | 44.1 | 51.8 |
| DACS[46] | 58.5 | 34.7 | 76.4 | 20.9 | 22.6 | 31.7 | 32.7 | 46.8 | 58.7 | 39 | 36.3 | 43.7 | 20.5 | 72.3 | 39.6 | 34.8 | 51.1 | 24.6 | 38.2 | 41.2 |
| MGCDA(RefineNet)[39] | 73.4 | 28.7 | 69.9 | 19.3 | 26.3 | 36.8 | 53 | 53.3 | 75.4 | 32 | 84.6 | 51 | 26.1 | 77.6 | 43.2 | 45.9 | 53.9 | 32.7 | 41.5 | 48.7 |
| DANNet(PSPNet)[51] | 84.3 | 54.2 | 77.6 | 38 | 30 | 18.9 | 41.6 | 35.2 | 71.3 | 39.4 | 86.6 | 48.7 | 29.2 | 76.2 | 41.6 | 43 | 58.6 | 32.6 | 43.9 | 50 |
| DANIA(PSPNet)[52] | 88.4 | 60.6 | 81.1 | 37.1 | 32.8 | 28.4 | 43.2 | 42.6 | 77.7 | 50.5 | 90.5 | 51.5 | 31.1 | 76 | 37.4 | 44.9 | 64 | 31.8 | 46.3 | 53.5 |
| ADVENT[49] | 72.9 | 14.3 | 40.5 | 16.6 | 21.2 | 9.3 | 17.4 | 21.2 | 63.8 | 23.8 | 18.3 | 32.6 | 19.5 | 69.5 | 36.2 | 34.5 | 46.2 | 26.9 | 36.1 | 32.7 |
| DAFormer(Baseline)[16] | 58.4 | 51.3 | 84 | 42.7 | 35.1 | 50.7 | 30 | 57 | 74.8 | 52.8 | 51.3 | 58.3 | 32.6 | 82.7 | 58.3 | 54.9 | 82.4 | 44.1 | 50.7 | 55.4 |
| STA(ours) | 82.11 | 42.98 | ↑**85.59** | 40.93 | 31.49 | ↑**51.66** | ↑**60.67** | ↑**57.92** | 75.97 | ↑**53.08** | 88.1 | ↑**61.99** | ↑**34.24** | ↑**82.87** | ↑**65.01** | ↑**63.36** | ↑**84.39** | ↑**44.58** | 49.57 | ↑**60.87** |

**Table 2: Quantitative comparison of STA with our baseline on Cityscapes to ACDC benchmarks(Fog, Night, Rain, and Snow). Both models are trained by STS.**

| Method | Road | S.walk | Build | Wall | Fence | Pole | Tr.Light | Sign | Veget | Terrain | Sky | Person | Rider | Car | Truck | Bus | Train | M.bike | Bike | mIou |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Cityscapes to ACDC-Fog | | | | | | | | | | | | | | | | | | | | |
| DAFormer[16] | 43.6 | 43.77 | 55.14 | 51.49 | 51.49 | 39.49 | 40.17 | 55.12 | 68.16 | 63.96 | 36.61 | 31.06 | 53.6 | 72.26 | 64.29 | 31.09 | 80.13 | 39.53 | 29.28 | 48.92 |
| STA(ours) | ↑**86.08** | ↑**53.46** | ↑**86.26** | 48.86 | 24.98 | ↑**45.86** | ↑**58.48** | ↑**56.05** | ↑**82.96** | 63.34 | ↑**97.5** | ↑**46.15** | ↑**57.61** | ↑**74.78** | 58.57 | ↑**38.01** | 76.95 | ↑**50.27** | ↑**37.44** | ↑**60.19** |
| Cityscapes to ACDC-Night | | | | | | | | | | | | | | | | | | | | |
| DAFormer | 74.81 | 58.63 | 72.7 | 30.56 | 19.76 | 38.67 | 15.7 | 37.16 | 49.11 | 43.32 | 45.17 | 56.93 | 25.37 | 68.71 | 14.29 | 40.2 | 82.67 | 30.55 | 44.12 | 44.65 |
| STA(ours) | ↑**89.42** | 56.17 | 71.06 | 30.37 | 17.7 | ↑**48.84** | ↑**32.78** | ↑**43.9** | 48.75 | 39.26 | 3.92 | ↑**58.33** | ↑**32.67** | ↑**74.68** | ↑**45.01** | ↑**63.46** | 76.74 | ↑**42.47** | 43.96 | ↑**48.39** |
| Cityscpaes to ACDC-Rain | | | | | | | | | | | | | | | | | | | | |
| DAFormer | 54.63 | 43.21 | 90.78 | 53.41 | 39.36 | 45.71 | 65.82 | 58.62 | 91.58 | 40.93 | 65.97 | 56.61 | 24 | 84.78 | 60.38 | 83.47 | 81.48 | 45.42 | 52.35 | 59.92 |
| STA(ours) | ↑**59.06** | 35.22 | ↑**92.12** | ↑**56.6** | 38.08 | ↑**50.93** | 65.29 | **63.32** | ↑**92.63** | 37.06 | ↑**74.31** | ↑**58.48** | ↑**26.04** | ↑**85.95** | ↑**62.76** | ↑**84.15** | 81.02 | 41.84 | ↑**58.93** | ↑**61.25** |
| Cityscpaes to ACDC-Snow | | | | | | | | | | | | | | | | | | | | |
| DAFormer | 45.69 | 33.38 | 85.41 | 35.55 | 40.08 | 46.15 | 58.5 | 37.17 | 87.9 | 4.04 | 54.89 | 64.35 | 42.52 | 83.07 | 61 | 67.6 | 82.75 | 33.32 | 56.62 | 53.68 |
| STA(ours) | ↑**54.62** | ↑**43.67** | ↑**85.44** | 35.47 | ↑**41.59** | ↑**51.85** | ↑**71.79** | ↑**62.07** | ↑**88.16** | ↑**9.54** | ↑**68.96** | 64.33 | 29.73 | ↑**87.2** | ↑**68.12** | ↑**75.25** | ↑**84.94** | 29.4 | 49.9 | ↑**58** |

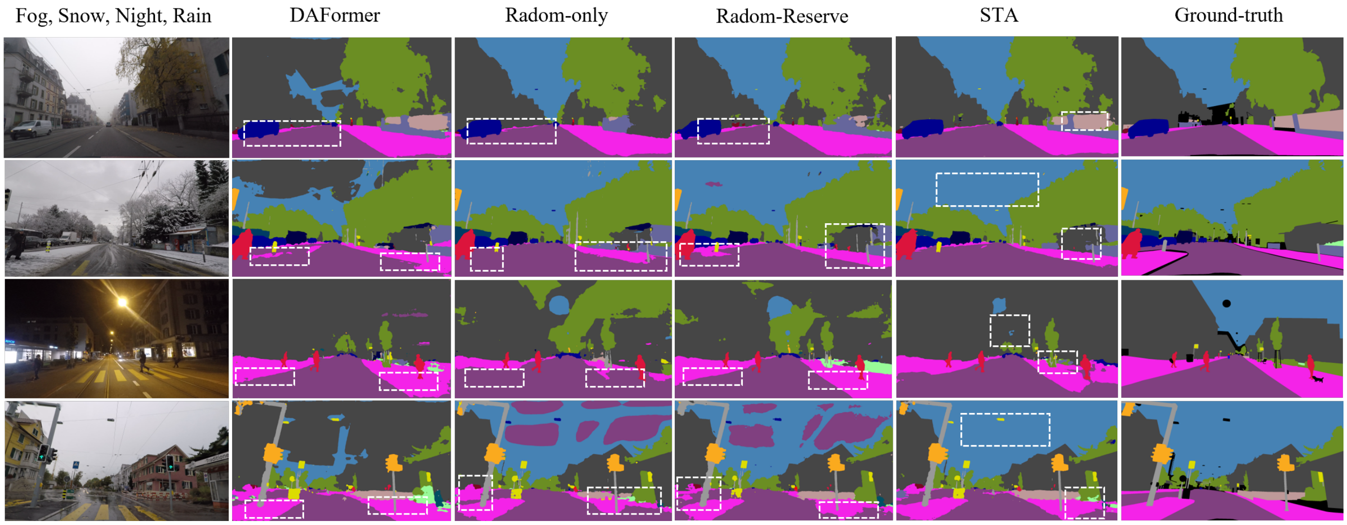

**Figure 6: The visual comparison between Random-only, Random-Reserve, STA, and DAFormer in every scene. We note that DAFormer, Random-only, and Random-Reserve are trained in STS, and STA is trained on the Foggy Zurich dataset.**

pixel-level annotation and the corresponding image is collected in almost the same scene under normal conditions.

**Foggy Zurich** [37] is a real-world foggy weather dataset captured in Zurich. It can be split into two parts, light, and medium, by the density of

**Table 3: Performance comparison to previous SOTA methods. Parameters of the Refusion module are $k$=2, $\lambda_{FZ1}$=0.01, $\lambda_{FZ2}$=0.06, $\lambda_{FD1}$=0.02, and $\lambda_{FD2}$=0.06.**

| Method | Backbone | FD-mIou | FZ-mIou |
|---|---|---|---|
| AdSegNet [48] | DeepLab-v2 | 37.6 | 26.1 |
| ADVENT [49] | DeepLab-v2 | 36.1 | 24.5 |
| DISE [4] | DeepLab-v2 | 45.2 | 40.7 |
| CCM [26] | DeepLab-v2 | 42.6 | 35.8 |
| SAC [1] | DeepLab-v2 | 43.4 | 37.0 |
| ProDA [55] | DeepLab-v2 | 41.2 | 37.8 |
| DACS [46] | DeepLab-v2 | 35.0 | 28.7 |
| CuDA-Net+ [30] | DeepLab-v2 | 53.5 | 49.1 |
| SP-FogAdapt+ [20] | ResNet-38 | 53.4 | 50.6 |
| FIFO [24] | RefineNet | 50.7 | 48.4 |
| SFSU [38] | RefineNet | 35.9 | 35.7 |
| CycleGAN [58] | RefineNet | 47.7 | 40.5 |
| MUNIT [25] | RefineNet | 47.8 | 39.1 |
| CMAda2 [37] | RefineNet | 37.3 | 42.9 |
| CMAda3+ [7] | RefineNet | 49.8 | 46.8 |
| DAFormer(baseline) [16] | SegFormer | 50.76 | 40.8 |
| **STA$^*$+ (ours)** | SegFormer | **54.85** | 46.9 |

[1] Methods followed by + mean that they use additional data during the training and * means that the STA model uses all modules and domains.

**Table 4: Ablation studies in all conditions of adverse scenarios to show the importance of the Reservation domain.**

| Methods | Components | mIou | | | | |
|---|---|---|---|---|---|---|
| | | **Fog** | **Night** | **Snow** | **Rain** | **All** |
| Init. | DAFormer | 48.92 | 44.65 | 53.68 | 59.92 | 55.36 |
| RM-only | STA | 56.21 | 44.90 | ↓53.06 | 61.01 | ↓55.25 |
| RD-only | STA | 50.69 | **51.34** | 55.47 | 60.48 | 57.64 |
| RM-RD | STA | ↑**58.3** | ↑49.65 | ↑**55.43** | ↑**61.46** | ↑**61.27** |

the fog. There is a total of 3808 images in Foggy Zurich. The light has 1522 images and the medium has 1498 images.

**Foggy Driving [37]** is a collection of 101 real-world foggy road scenarios, with semantic segmentation and object detection annotations, used as a test benchmark for the foggy scene tasks.

**Dark Zurich [8]** is a collection of 8779 images captured at nighttime, twilight, and daytime, along with the respective GPS coordinates of the camera for each image. There are 50 validation, and 151 test images for nighttime in it.

**Clear Zurich** consists of 1498 clear weather images which are selected randomly from ACDC (fog-ref, night-ref, rain-ref, snow-ref). We use this dataset to be our Reservation domain in later experiments.

## 4.2 STS Performance Comparison

In the Introduction, we argue that the third assumption of achieving Single to All is to demonstrate the ability of STA on every ACDC benchmark using the STS strategy. To that end, we compare STA with other previous excellent works on all ACDC benchmarks, Foggy Zurich, and Foggy Driving to demonstrate its versatility across different target domains. All methods are trained using the STS strategy. And all experiments are conducted on an NVIDIA 32GB V100 GPU and we follow the training strategy of [16].

**Cityscapes to ACDC** In these experiments, we conduct using the Cityscapes images as the source domain and the adverse scenarios images from ACDC as the target domain. Additionally, clear weather images (1600ACDC-ref) corresponding to the target domains are reserved as the Reservation domain. We test STA on all conditions benchmark, and results presented in Table 1 demonstrate that STA achieves a score of 60.87 in all conditions benchmark, outperforming our baseline DAFormer 5.47.

To make a more accurate comparison, we respectively conduct experiments on Fog, Night, Snow, and Rain benchmarks. As shown in Table 2, STA outperforms both our baseline and one of the inspirations in every benchmark, with the best result achieved on the fog benchmark, where STA obtains a score of 60.19, outperforming our baseline of 11.27. For the fog benchmark, we use the Refusion module with $k = 1$, $\lambda_{fog1} = 0.01$, while for the rain, snow, and night, we employ the two-layer module with $\lambda_{rain1} = 0.01, \lambda_{rain2} = 0.8, \lambda_{snow1} = 0.01, \lambda_{snow2} = 0.3, \lambda_{night1} = 0.01$ and $\lambda_{night2} = 0.01$.

**Cityscapes to Foggy Zurich and Foggy Driving** In the Cityscapes to Foggy Zurich experiments, we construct the target domain using foggy images (1498medium + 802light) from the Foggy Zurich dataset, while the RD consists of clear Zurich and foggy images (1498ACDC-ref + 802light). For Cityscapes to Foggy Driving, we use target domain images (1498medium) from Foggy Zurich datasets and RD images from Clear Zurich. Table 3 shows that our results achieved SOTA performance on Foggy Driving and outstanding performance on Foggy Zurich, outperforming our baseline by 6.1. Additionally, Fig 5 shows the influence of different parameters of the Refusion module on the Foggy STA benchmark.

## 4.3 Scientific Assumption Validation

We begin by recapping our assumptions: (1) RM is essential, as it can significantly improve the mutual similarity of factors to unify different scenes. (2) RD, reserving the style factors, is also essential, as the Randomization module may lose the style factors due to the strong style transform. (3) Based on the unification by RM and RD, the STA model can perform well on every benchmark by STS. In the above STS experiments, we prove the third assumption that STA is competent in STS training. And in this section, we are going to validate the first and second assumptions by demonstrating the visualization and quantitative results of ablation studies between RD and RM.

**Ablation studies of RM and RD** Ablation studies and visualizations of RM and RD will be conducted to further illustrate how they work together. We will perform ablation experiments with three flows: RM-only, RD-only, and RM-RD, respectively. We will use the same training datasets, test sets, and STS strategy without the Refusion module to avoid other interferences. Specifically, we will set the four adverse scene images from the ACDC dataset as four target domains (400 images) and the clear weather Cityscapes images as the source domain. The difference between the two flows is the usage of RD, consisting of clear weather images from ACDC-ref (400 images), while RM-only does not purposely reserve the style factors and RD-only does not learn the unified factors. The comparison of these two flows is shown in Table 4.

We observe that the results of RM-only and RD-only are lower than those of RM-RD in every scene benchmark. Especially in the snow, night, and all-condition scenes, the performance of RM-only falls even below that of DAFormer. This suggests that the loss of style factors can have a negative effect on performance in some scenes. On the other hand, the results of RD-only are much lower than RM-only under the fog scene. Although the result of RD-only is significant in the night scene, the results under other scenes are too low to generalize. On the contrary, after introducing the Reservation domain to build RM-RD, the results significantly improved under every scene, especially in the all-conditions benchmark, where it outperforms RM-only by over 6 mIou. These results show that the Reservation domain can improve the robustness of STA by facilitating the learning of additional knowledge about style factors.

**Visualization of RM and RD** To further validate our assumptions, we use color predictions to visualize the loss and reservation of style factors, as

**Table 5: Ablation studies show the domain generalizability of STA.**

| Comparison | Method | Target domain | | | | | Test Adverse Senarios | | | | Domain Generalization Gain | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ACDC-Fog | ACDC-Rain | ACDC-Snow | Foggy Zurich | Dark Zurich | Fog | Night | Rain | Snow | Tar-gain | Unseen-gain | Avg-gain |
| Base | DAFormer(STS) | | | | | | 48.92 | 44.65 | 59.92 | 53.68 | 0.00 | 0.00 | 0.00 |
| | DAFormer | ✓ | | | | | 48.92 | ↓39.56 | ↓57.40 | ↓52.10 | 0.00 | -9.19 | -3.06 |
| | STA$^*$ | ✓ | | | | | ↑**60.19** | 44.51 | ↑**63.72** | ↑**60.51** | **+11.27** | **+10.49** | **+5.44** |
| | DAFormer | | ✓ | | | | ↑51.08 | ↓41.79 | 59.92 | ↓52.27 | 0.00 | -2.11 | -0.7 |
| | STA$^*$ | | ✓ | | | | ↑52.37 | ↓43.82 | ↑**63.72** | ↑56.09 | **+3.8** | **+5.03** | **+2.21** |
| | DAFormer | | | ✓ | | | ↑51.30 | ↓40.75 | ↑60.90 | ↑53.68 | 0.00 | +0.6 | +0.2 |
| | STA$^*$ | | | ✓ | | | ↑55.40 | ↑45.65 | ↑61.97 | ↑58.00 | **+4.32** | **+9.53** | **+3.46** |
| FZ | DAFormer+ | | | | ✓ | | 49.6 | 44.9 | 58.16 | 52.28 | 0.00 | 0.00 | 0.00 |
| | STA+ | | | | ✓ | | ↑60.04 | ↑**47.57** | ↑65.13 | ↑60.54 | +10.44 | +17.60 | +7.09 |
| | STA$^*$+ | | | | ✓ | | ↑**60.86** | ↑46.11 | ↑**65.65** | ↑**63.35** | **+11.26** | **+19.77** | **+7.77** |
| DZ | DAFormer+ | | | | | ✓ | 57.81 | **52.75** | 61.68 | 57.3 | 0.00 | 0.00 | 0.00 |
| | STA+ | | | | | ✓ | ↑60.24 | ↓50.99 | ↑63.03 | ↑60.29 | -1.76 | +6.77 | +1.25 |
| | STA$^*$+ | | | | | ✓ | ↑**61.37** | ↓50.19 | ↑**63.66** | ↑**62.23** | -2.56 | **+10.47** | **+1.98** |

shown in Fig. 6. The observed results align well with our ablation studies. As previously mentioned, we defined the style factors as empirically concretized to the roads and sidewalks in a city scene. Therefore, we use the visualization of sidewalks and roads to show the gain and loss of style factors.

It is apparent that the RM-only flow loses some information about the sidewalk and road classes in every scene compared with the baseline, and then the RM-RD learns this information back almost in the same position, which is the labels framed by the white dotted box. RM-RD demonstrates a better ability to segment the classes related to style factors than RM-only in each adverse scenario. The visualized results further support our first two assumptions: Both RM and RD Matter.

**Cross-scene Generalization Experiments** Based on the validation above, we have completed the preparations for achieving Single to All and provided sufficient interpretability of our methods. Therefore, we will now validate the Single-to-All ability of our STA model.

First, we introduce the settings for the ablation studies in Fig 5. The comparisons are divided into three groups: Base, FZ, and DZ. All comparisons are tested on the same benchmarks (ACDC-fog, night, rain, and snow), with the main distinction being the different training datasets used for the target domains. DAFormer (STS) serves as the benchmark for comparison, showing the upper-bound performance of DAFormer on these benchmarks using the STS training strategy. Other methods use single-scene datasets for training and testing on all conditions of ACDC. In Base, we set DAFormer (STS) as the reference for comparison with DAFormer and STA$^*$. In FZ and DZ, we leverage DAFormer+ as the reference. To evaluate the generalizability of STA, we introduce three types of gains: Tar-gain, Unseen-gain, and Total-gain. Tar-gain corresponds to the gain of models in the target domain. For example, in FZ comparisons, we train baseline and STA on Foggy Zurich dataset. Thus Tar-gain represents the variation in mIou of STA+ and STA$^*$+ compared with DAFormer+ on the fog scene test. Similarly, Tar-gain for DZ represents the gains on the nighttime scene test. Unseen-gain measures the variation under the unseen domain of STA. For FZ, we calculate the cumulative gain in mIou of the night, rain, and snow scenes instead of the fog scene to obtain Unseen-gain. Avg-gain is the mean variation in mIou of every scene.

For every method, STA$^*$ means STA utilizing all modules and domains, STA+ means STA trained on additional datasets without the Refusion module, and STA$^*$+ means STA trained with all modules, domains, and extra data.

**Performance Analysis** In the Base comparison, it is evident that STA$^*$ demonstrates domain generalizability to all conditions, achieving positive gains by training on fog, rain, and snow scenes. On the contrary, DAFormer, which employs the same strategy as STA$^*$, shows almost no performance of domain generalization, with negative gains under fog and rain training. Although DAFormer achieves a few positive gains by training on some scenes, the results do not demonstrate generalizability.

To eliminate the influence of the ACDC datasets and further investigate the generalization of STA, we train STA and DAFormer on two other datasets under different scenes. In FZ, STA+ still demonstrates good generalization in every scene with 10.44 Tar-gain, 17.60 Unseen-gain, and 7.09 Avg-gain. STA$^*$+ obtains better gains than STA+ by using the Refusion module with $k = 2$, $\lambda_1 = 0.01$, and $\lambda_2 = 0.06$. The FZ comparison demonstrates the strong domain generalizability of STA. However, we still argue that a single FZ is not enough to validate our assumption that STA can generalize to all conditions from a single scene. Thus, we perform the DZ comparison to further investigate the generalizability of STA, as DZ consists of night scene images, which is the toughest scene in current UDA tasks.

In DZ, we train STA on the night Zurich dataset and evaluate its performance on other adverse scene tests to demonstrate its generalization. In Table 5, STA+ and STA$^*$+ outperform the baseline, and STA$^*$+ obtains 10.47 Unseen-gain and 1.98 Avg-gain overall, demonstrating that STA generalizes well to all conditions, even with training on the night scene. The parameters of STA$^*$+ in DZ are $k = 2$, $\lambda_1 = 0.01$, and $\lambda_2 = 0.01$.

These extensive experiments demonstrate that STA indeed has wide domain generalizability in all adverse conditions on the ACDC benchmarks. To showcase the single-to-all generalizability of STA as accurately as possible, we use almost all the common UDA adverse scenario datasets, including the fog, night, rain, and snow scenes. Thus, we draw a scientifically solid conclusion that STA achieves Single to All in UDA for semantic segmentation tasks by adapting from a single-scene dataset to all adverse conditions on the ACDC benchmarks.

## 5 CONCLUSION

In this paper, we analyze the current problems in UDA tasks, meticulously introduce the coming of our ideas, and carefully validate our scientific assumption through a series of ablation experiments and visual presentations. We propose the STA model to achieve Single to All in UDA for semantic segmentation tasks. Specifically, we build the STA model to train on a single scene dataset to adapt to all conditions of ACDC adverse benchmarks. The performance of STA outperforms current state-of-the-art methods in the Foggy Driving benchmark and achieves outstanding results on other authoritative benchmarks. We will make the code publicly available at https://github.com/Cuzyoung/STA.

**Limitation** Although STA has domain generalizability, the limitation remains that the Refusion module elevates the overall generalizability by slightly decreasing the performance in night-scene tasks as shown in Table 5. Therefore, efforts to further improve the ability of STA to adapt to the night scene are a future direction to explore.

**Acknowledge** This study is supported by Innovation Group Project of Southern Marine Science and Engineering Guangdong Laboratory (Zhuhai) (No. 311022006).

# REFERENCES

[1] Nikita Araslanov and Stefan Roth. 2021. Self-supervised augmentation consistency for adapting semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 15384–15394.

[2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. arXiv preprint arXiv:1907.02893 (2019).

[3] David Brüggemann, Christos Sakaridis, Prune Truong, and Luc Van Gool. 2023. Refign: Align and refine for adaptation of semantic segmentation to adverse conditions. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 3174–3184.

[4] Wei-Lun Chang, Hui-Po Wang, Wen-Hsiao Peng, and Wei-Chen Chiu. 2019. All about structure: Adapting structural information across domains for boosting semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 1900–1909.

[5] Yuhua Chen, Wen Li, Xiaoran Chen, and Luc Van Gool. 2019. Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 1841–1850.

[6] 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.

[7] Dengxin Dai, Christos Sakaridis, Simon Hecker, and Luc Van Gool. 2020. Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding. International Journal of Computer Vision 128 (2020), 1182–1204.

[8] Dengxin Dai and Luc Van Gool. 2018. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In 2018 21st International Conference on Intelligent Transportation Systems (ITSC). IEEE, 3819–3824.

[9] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. 2016. A learned representation for artistic style. arXiv preprint arXiv:1610.07629 (2016).

[10] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. The journal of machine learning research 17, 1 (2016), 2096–2030.

[11] Huan Gao, Jichang Guo, Guoli Wang, and Qian Zhang. 2022. Cross-domain correlation distillation for unsupervised domain adaptation in nighttime semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 9913–9923.

[12] Golnaz Ghiasi, Honglak Lee, Manjunath Kudlur, Vincent Dumoulin, and Jonathon Shlens. 2017. Exploring the structure of a real-time, arbitrary neural artistic stylization network. arXiv preprint arXiv:1705.06830 (2017).

[13] Tejas Gokhale, Rushil Anirudh, Jayaraman J Thiagarajan, Bhavya Kailkhura, Chitta Baral, and Yezhou Yang. 2023. Improving Diversity with Adversarially Learned Transformations for Domain Generalization. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 434–443.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. Commun. ACM 63, 11 (2020), 139–144.

[15] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. 2018. Cycada: Cycle-consistent adversarial domain adaptation. In International conference on machine learning. Pmlr, 1989–1998.

[16] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. 2022. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 9924–9935.

[17] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. 2022. HRDA: Context-aware high-resolution domain-adaptive semantic segmentation. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXX. Springer, 372–391.

[18] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. 2022. MIC: Masked Image Consistency for Context-Enhanced Domain Adaptation. arXiv preprint arXiv:2212.01322 (2022).

[19] Xun Huang and Serge Belongie. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the IEEE international conference on computer vision. 1501–1510.

[20] Javed Iqbal, Rehan Hafiz, and Mohsen Ali. 2022. FogAdapt: Self-supervised domain adaptation for semantic segmentation of foggy images. Neurocomputing 501 (2022), 844–856.

[21] Philip TG Jackson, Amir Atapour Abarghouei, Stephen Bonner, Toby P Breckon, and Boguslaw Obara. 2019. Style augmentation: data augmentation via style randomization.. In CVPR workshops, Vol. 6. 10–11.

[22] Christoph Kamann and Carsten Rother. 2020. Benchmarking the robustness of semantic segmentation models. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 8828–8838.

[23] Dong-Hyun Lee et al. [n. d.]. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks.

[24] Sohyun Lee, Taeyoung Son, and Suha Kwak. 2022. Fifo: Learning fog-invariant features for foggy scene segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 18911–18921.

[25] Yayun Lei, Takanori Emaru, Ankit A Ravankar, Yukinori Kobayashi, and Su Wang. 2020. Semantic image segmentation on snow driving scenarios. In 2020 IEEE International Conference on Mechatronics and Automation (ICMA). IEEE, 1094–1100.

[26] Guangrui Li, Guoliang Kang, Wu Liu, Yunchao Wei, and Yi Yang. 2020. Content-consistent matching for domain adaptive semantic segmentation. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV. Springer, 440–456.

[27] Wei Li, Kai Liu, Lin Yan, Fei Cheng, YunQiu Lv, and LiZhe Zhang. 2019. FRD-CNN: Object detection based on small-scale convolutional neural networks and feature reuse. Scientific reports 9, 1 (2019), 1–12.

[28] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. 2019. Bidirectional learning for domain adaptation of semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 6936–6945.

[29] Yinghong Liao, Wending Zhou, Xu Yan, Shuguang Cui, Yizhou Yu, and Zhen Li. 2022. Geometry-Aware Network for Domain Adaptive Semantic Segmentation. arXiv preprint arXiv:2212.00920 (2022).

[30] Xianzheng Ma, Zhixiang Wang, Yacheng Zhan, Yinqiang Zheng, Zheng Wang, Dengxin Dai, and Chia-Wen Lin. 2022. Both style and fog matter: Cumulative domain adaptation for semantic foggy scene understanding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 18922–18931.

[31] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. 2020. Instance adaptive self-training for unsupervised domain adaptation. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16. Springer, 415–430.

[32] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. 2021. Classmix: Segmentation-based data augmentation for semi-supervised learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 1369–1378.

[33] Fengchun Qiao, Long Zhao, and Xi Peng. 2020. Learning to learn single domain generalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 12556–12565.

[34] Nikhil Reddy, Abhinav Singhal, Abhishek Kumar, Mahsa Baktashmotlagh, and Chetan Arora. 2022. Master of all: Simultaneous generalization of urban-scene segmentation to all adverse weather conditions. In European Conference on Computer Vision. Springer, 51–69.

[35] Fuji Ren, Wenjie Liu, and Guoqing Wu. 2019. Feature reuse residual networks for insect pest recognition. IEEE access 7 (2019), 122758–122768.

[36] Eduardo Romera, Luis M Bergasa, Kailun Yang, Jose M Alvarez, and Rafael Barea. 2019. Bridging the day and night domain gap for semantic segmentation. In 2019 IEEE Intelligent Vehicles Symposium (IV). IEEE, 1312–1318.

[37] Christos Sakaridis, Dengxin Dai, Simon Hecker, and Luc Van Gool. 2018. Model adaptation with synthetic and real data for semantic dense foggy scene understanding. In Proceedings of the european conference on computer vision (ECCV). 687–704.

[38] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. 2018. Semantic foggy scene understanding with synthetic data. International Journal of Computer Vision 126 (2018), 973–992.

[39] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. 2020. Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 44, 6 (2020), 3139–3153.

[40] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. 2021. ACDC: The adverse conditions dataset with correspondences for semantic driving scene understanding. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 10765–10775.

[41] Swami Sankaranarayanan and Yogesh Balaji. 2023. Meta learning for domain generalization. In Meta-Learning with Medical Imaging and Health Informatics Applications. Elsevier, 75–86.

[42] Lei Sun, Kaiwei Wang, Kailun Yang, and Kaite Xiang. 2019. See clearer at night: towards robust nighttime semantic segmentation through day-night image conversion. In Artificial Intelligence and Machine Learning in Defense Applications, Vol. 11169. SPIE, 77–89.

[43] Tao Sun, Cheng Lu, Tianshuo Zhang, and Haibin Ling. 2022. Safe self-refinement for transformer-based domain adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 7191–7200.

[44] Zhiqiang Tang, Yunhe Gao, Yi Zhu, Zhi Zhang, Mu Li, and Dimitris N Metaxas. 2021. Selfnorm and crossnorm for out-of-distribution robustness. (2021).

[45] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. Advances in neural information processing systems 30 (2017).

[46] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. 2021. Dacs: Domain adaptation via cross-domain mixed sampling. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 1379–1389.

[47] Maxime Tremblay, Shirsendu Sukanta Halder, Raoul De Charette, and Jean-François Lalonde. 2021. Rain rendering for evaluating and improving robustness to bad weather. International Journal of Computer Vision 129 (2021), 341–360.

[48] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. 2018. Learning to adapt structured output space for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition. 7472–7481.

[49] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. 2019. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2517–2526.

[50] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. 2019. Dada: Depth-aware domain adaptation in semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 7364–7373.

[51] Xinyi Wu, Zhenyao Wu, Hao Guo, Lili Ju, and Song Wang. 2021. Dannet: A one-stage domain adaptation network for unsupervised nighttime semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 15769–15778.

[52] Xinyi Wu, Zhenyao Wu, Lili Ju, and Song Wang. 2021. A one-stage domain adaptation network with image alignment for unsupervised nighttime semantic segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 45, 1 (2021), 58–72.

[53] Yanchao Yang and Stefano Soatto. 2020. Fda: Fourier domain adaptation for semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 4085–4095.

[54] Jan-Nico Zaech, Dengxin Dai, Martin Hahner, and Luc Van Gool. 2019. Texture underfitting for domain adaptation. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC). IEEE, 547–552.

[55] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. 2021. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 12414–12424.

[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] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. 2021. Domain generalization with mixstyle. arXiv preprint arXiv:2104.02008 (2021).

[58] 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.