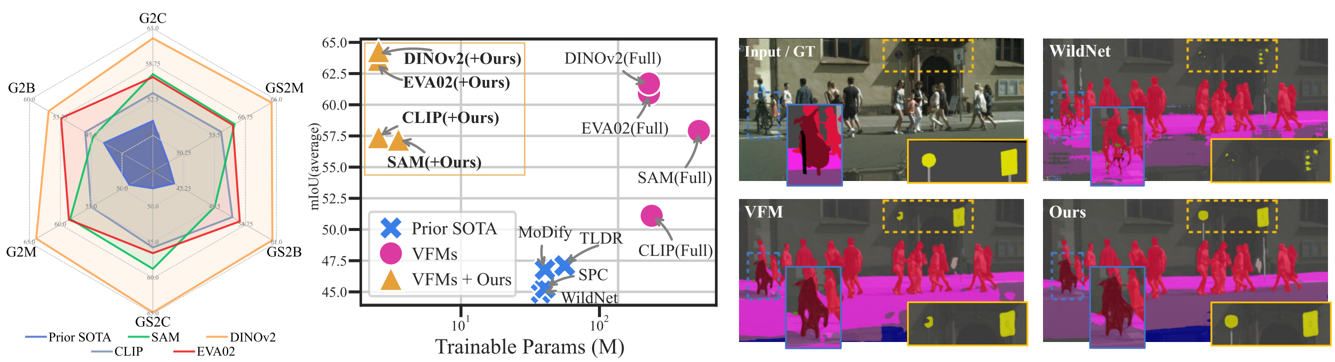
# Stronger, Fewer, & Superior: Harnessing Vision Foundation Models for Domain Generalized Semantic Segmentation

Zhixiang Wei[1*] Lin Chen[1,2*] Yi Jin[1*] Xiaoxiao Ma[1] Tianle Liu[1] Pengyang Ling[1,2] Ben Wang[1]
Huaian Chen[1†] Jinjin Zheng[1]

[1] University of Science and Technology of China [2] Shanghai AI Laboratory

{zhixiangwei,chlin,xiao_xiao,tleliu,lpyang27,wblzgrsn,anchen}@mail.ustc.edu.cn
{jinyi08,jjzheng}@ustc.edu.cn




(a) **Stronger** pre-trained models (b) **Fewer** trainable parameters (c) **Superior** generalization ability

Figure 1. Vision Foundation Models (VFMs) are **stronger** pre-trained models that serve as robust backbones, effortlessly outperforming previous state-of-the-art Domain Generalized Semantic Segmentation (DGSS), as shown in (a). Yet, the extensive parameters of VFMs make them challenging to train. To address this, we introduce a robust fine-tuning approach to efficiently harness VFMs for DGSS. As illustrated in (b) and (c), the proposed methods achieve **superior** generalizability with **fewer** trainable parameters within backbones.

## Abstract

*In this paper, we first assess and harness various Vision Foundation Models (VFMs) in the context of Domain Generalized Semantic Segmentation (DGSS). Driven by the motivation that* ***Leveraging Stronger pre-trained models and Fewer trainable parameters for Superior generalizability****, we introduce a robust fine-tuning approach, namely "*__*Rein*__*", to parameter-efficiently harness VFMs for DGSS. Built upon a set of trainable tokens, each linked to distinct instances, Rein precisely refines and forwards the feature maps from each layer to the next layer within the backbone. This process produces diverse refinements for different categories within a single image. With fewer trainable parameters, Rein efficiently fine-tunes VFMs for DGSS tasks, surprisingly surpassing full parameter fine-tuning. Extensive experiments across various settings demonstrate that Rein significantly outperforms state-of-the-art methods. Remarkably, with just an extra* ***1%*** *of trainable parameters within the frozen backbone, Rein achieves a mIoU of* ***78.4%*** *on the Cityscapes, without accessing any real urban-scene datasets. Code is available at* https://github.com/w1oves/Rein.git.

## 1. Introduction

Prior works [32, 34, 36, 56, 63, 66, 72] in Domain Generalized Semantic Segmentation (DGSS) focus on improving prediction accuracy across multiple unseen domains without accessing their data, thus enabling a high generalization for real applications. Since models are fine-tuned using datasets [11, 57] that are either limited in scale or different in image style from the target domain, complex data augmentation approaches [4, 53, 73] and domain invariant feature extraction strategies [9, 51, 62, 67] have been widely explored in previous DGSS. These methods result in enhanced generalization when applied to classic backbones, *e.g.*, VGGNet [61], MobileNetV2 [60], and ResNet [23].

In recent years, large-scale Vision Foundation Models (VFMs) like CLIP [55], MAE [24], SAM [38], EVA02 [17, 18], and DINOv2 [50] have significantly advanced the

* indicates equal contributions.
† Corresponding authors.

|  | Previous DGSS methods |  |  |  |  |  | Frozen backbone of VFMs |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Methods | GTR[53] | AdvStyle[73] | WildNet[40] | SPC[27] | PASTA[4] | TLDR[37] | CLIP-ViT-L[55] | MAE-L[24] | SAM-H[38] | EVA02-L[17] | DINOv2-L[50] |
| Publications | TIP21 | NIPS22 | CVPR22 | CVPR23 | ICCV23 | ICCV23 | ICML21 | CVPR22 | ICCV23 | arXiv23 | arXiv23 |
| mIoU (Citys) | 43.7 | 43.4 | 45.8 | 46.7 | 45.3 | 47.6 | 53.7 | 43.3 | 57.0 | 56.5 | **63.3** |
| mIoU (BDD) | 39.6 | 40.3 | 41.7 | 43.7 | 42.3 | 44.9 | 48.7 | 37.8 | 47.1 | 53.6 | **56.1** |
| mIoU (Map) | 39.1 | 42.0 | 47.1 | 45.5 | 48.6 | 48.8 | 55.0 | 48.0 | 58.4 | 58.6 | **63.9** |
| mIoU (Average) | 40.8 | 41.9 | 44.9 | 45.3 | 45.4 | 47.1 | 52.4 | 43.0 | 54.2 | 56.2 | **61.1** |

Table 1. Performance benchmarking of **multiple VFMs and previous DGSS methods** under the *GTAV → Cityscapes (Citys) + BDD100K (BDD) + Mapillary (Map)* generalization setting. Without specialized design, frozen VFMs demonstrate **stronger** performance.

boundaries of performance in a variety of computer vision challenges. Giving the remarkable generalization of these VFMs across various unseen scenes, two intuitive questions emerge: *How do VFMs perform in the context of DGSS?* And *How to harness VFMs for DGSS?* We attempt to answer these questions as follows:

**Stronger**: We begin by evaluating and comparing the performance of various VFMs against existing DGSS methods. To ensure a fair comparison, we use image encoders from a variety of VFMs as the backbone for feature extraction in all cases. These backbones are coupled with the widely-used decode head, *i.e.*, Mask2Former [8], to generate semantic predictions. As illustrated in Tab. 1, while previous DGSS methods have showcased commendable results, they perform less effectively compared to frozen VFMs. This finding clearly demonstrates the powerful potential of VFMs in DGSS, outperforming traditional backbones like ResNet [23] and MobileNetV2 [60], thereby establishing VFMs as a meaningful benchmark in the field.

**Fewer**: Although VFMs have exhibited impressive generalization capabilities, fine-tuning them for DGSS tasks poses a challenge. The datasets [11, 57] commonly used in DGSS tasks are significantly smaller in scale compared to ImageNet [12], and fine-tuning VFMs with their huge number of trainable parameters on these datasets result in limited generalizability [33]. To address this issue, instead of the difficult task of large datasets collection, we resort to fine-tuning VFMs with fewer trainable parameters. However, most existing parameter-efficient fine-tuning strategies, which fine-tune a large-scale model with fewer trainable parameters, are primarily designed for adapting large language models [25, 26, 41, 43, 44, 70, 74] or classification networks [6, 28]. These methods are not developed for refining features for distinct instances within a single image, thereby limiting their effectiveness in DGSS contexts.

**Superior**: In this work, we introduce a robust and efficient fine-tuning approach, namely "Rein". Tailored for DGSS tasks, Rein employs fewer trainable parameters to harness stronger VFMs for achieving superior generalization. At its core, Rein comprises a set of randomly initialized tokens, each directly linked to different instances. These tokens, through a dot-product operation with VFMs features, generate an attention-like similarity map. This map enables Rein to perform precise refinement tailored to each instance within an image, significantly boosting VFMs in the context of DGSS. Moreover, to reduce the number of trainable parameters, we employ shared weights across MLPs in different layers and design our learnable tokens by multiplying two low-rank matrices. Extensive experiments on various DGSS settings demonstrate that the proposed Rein outperforms existing DGSS methods by a large margin with fewer trainable parameters. In a nutshell, the **main contributions** of this paper are as follows:

- We first assess various Vision Foundation Models (VFMs) in the context of Domain Generalized Semantic Segmentation (DGSS). Our extensive experiments in the DGSS framework highlight the impressive generalization capabilities of VFMs. The findings confirm that VFMs serve as **Stronger** backbones, thereby establishing a significant benchmark in this field.
- We present a robust fine-tuning method, namely "**Rein**", to parameter-efficiently harness VFMs. At its core, Rein consists of a set of learnable tokens, each directly linked to instances. With deliberate design, this linkage enables Rein to refine features at an instance-level within each layer. As a result, Rein reinforces the ability of VFMs in DGSS tasks, achieving this with **Fewer** trainable parameters while preserving the pre-trained knowledge.
- Comprehensive experiments across various DGSS settings demonstrate that Rein employs **Fewer** trainable parameters to effectively leverage **Stronger** VFMs for achieving **Superior** generalizability. This performance surpasses existing DGSS methods by a large margin. Notably, Rein is designed to integrate smoothly with existing plain vision transformers, improving their generalization ability and making training more efficient.

## 2. Related Works

**Domain Generalized Semantic Segmentation.** Domain Generalized Semantic Segmentation (DGSS) focuses on enhancing model generalizability. This field involves training models on a set of source domain to enhance their performance on distinct and unseen target domain. Various approaches [5, 13, 20, 21, 27, 29–31, 42, 54, 65] have been proposed to address this issue, with methods including splitting the learned features into domain-invariant and domain-specific components [62, 67], or employing meta-learning to train more robust models [35]. A standard scenario in

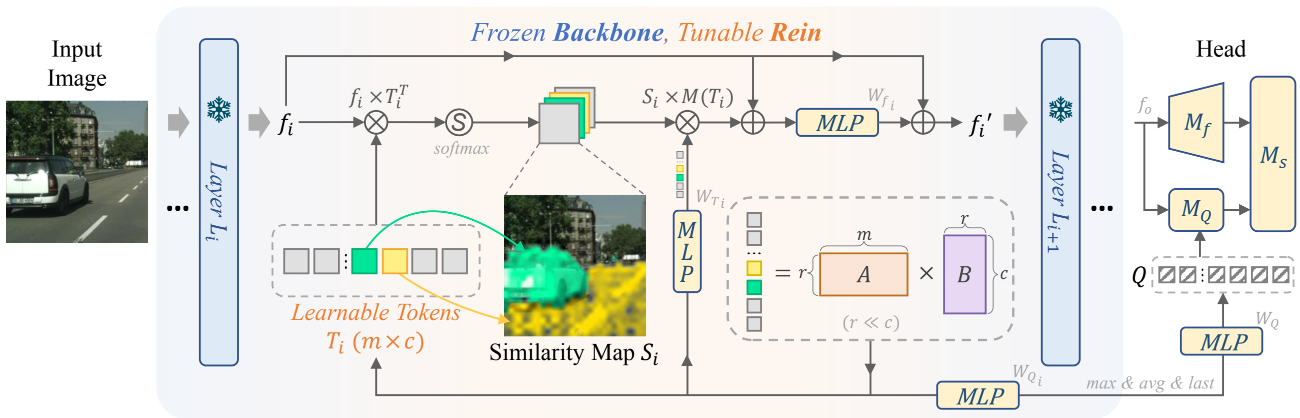



Figure 2. An overview of proposed Rein. Rein primarily consists of a collection of low-rank learnable tokens, denoted as $T = \{T_1, T_2, \ldots, T_N\}$. These tokens establish direct connections to distinct instances, facilitating instance-level feature refinement. This mechanism results in the generation of an enhancement feature map $f_i' = f_i + Rein(f_i)$ for each layer within backbone. All parameters of MLPs are layer-shared to reduce the number of parameters. $M_f$, $M_Q$, and $M_S$ are features module, queries module, and segmentation module, respectively. The notation $max$ & $avg$ & $last$ refers to the equation Eq. (8) and Eq. (10).

| Backbone | Fine-tune Method | Trainable Params* | mIoU | | | |
|---|---|---|---|---|---|---|
| | | | Citys | BDD | Map | Avg. |
| CLIP [55] (ViT-Large) | Full | 304.15M | 51.3 | 47.6 | 54.3 | 51.1 |
| | Freeze | 0.00M | 53.7 | 48.7 | 55.0 | 52.4 |
| | Rein | 2.99M | **57.1** | **54.7** | **60.5** | **57.4** |
| MAE [24] (Large) | Full | 330.94M | 53.7 | **50.8** | 58.1 | 54.2 |
| | Freeze | 0.00M | 43.3 | 37.8 | 48.0 | 43.0 |
| | Rein | 2.99M | **55.0** | 49.3 | **58.6** | **54.3** |
| SAM [38] (Huge) | Full | 632.18M | 57.6 | 51.7 | 61.5 | 56.9 |
| | Freeze | 0.00M | 57.0 | 47.1 | 58.4 | 54.2 |
| | Rein | 4.51M | **59.6** | **52.0** | **62.1** | **57.9** |
| EVA02 [17, 18] (Large) | Full | 304.24M | 62.1 | 56.2 | 64.6 | 60.9 |
| | Freeze | 0.00M | 56.5 | 53.6 | 58.6 | 56.2 |
| | Rein | 2.99M | **65.3** | **60.5** | **64.9** | **63.6** |
| DINOV2 [50] (Large) | Full | 304.20M | 63.7 | 57.4 | 64.2 | 61.7 |
| | Freeze | 0.00M | 63.3 | 56.1 | 63.9 | 61.1 |
| | Rein | 2.99M | **66.4** | **60.4** | **66.1** | **64.3** |

Table 2. Performance Comparison with the proposed **Rein across Multiple VFMs** as Backbones under the *GTAV* $\rightarrow$ *Cityscapes (Citys) + BDD100K (BDD) + Mapillary (Map)* generalization setting. Mark $*$ denotes trainable parameters in backbones.

DGSS is generalizing from one urban-scene dataset to another, for instance, from the synthetic GTAV [57] dataset to the real-world Cityscapes [11]. In this classic setting, certain techniques [9, 51, 52] have achieved notable performance through learning feature normalization/whitening schemes, while others [40] have improved segmentation results through feature-level style transfer and the introduction of additional data. Additionally, strong data augmentation [4, 16, 53, 73] often simply and effectively enhances model robustness. However, most of previous DGSS methods generally utilize outdated backbones like ResNet [23], VGGNet [61], MobileNetV2 [60], and ShuffleNetV2 [47], thereby leaving the efficacy of stronger Vision Foundation Models (VFMs) in DGSS relatively unexplored.

**Vision Foundation Models.** The concept of a Foundation Model, initially introduced by Bommasani *et al.* [2] in the field of Natural Language Processing (NLP), defined as "the base models trained on large-scale data in a self-supervised or semi-supervised manner that can be adapted for several other downstream tasks". While models like the ViT [14] and Swin Transformer [45] have demonstrated excellent performance, the quest for a Vision Foundation Model (VFM) similar to their NLP counterparts is ongoing. This pursuit has yielded significant advancements with the advent of models such as CLIP [55], which learn high-quality visual representation by exploring contrastive learning with large-scale image text pairs; MAE [24], utilizing a masked image modeling for learning latent image representations; SAM [38], which develops a promptable model and pre-train it on a broad dataset for segmentation task; EVA02 [17, 18], which integrates Masked Image Modeling pre-training with CLIP's vision features; and DINOv2 [50], which is pretrained on extensive, curated datasets without explicit supervision. These VFMs have shown remarkable performance in downstream applications. Yet, a dedicated investigation into their performance in the specific context of DGSS tasks remains unexplored.

**Parameter-Efficient Fine-tuning.** In the realm of NLP, parameter-efficient fine-tuning (PEFT) has achieved notable success by freezing most parameters of VFMs and fine-tuning a select few. Various approaches have been developed, such as BitFit [70], which adjusts only the model's bias terms; Prompt-tuning [41], introducing soft prompts to adapt frozen language models; Adapter-tuning [25], adding lightweight modules to each transformer layer; and notably, LoRA [26], which injects trainable rank decomposition matrices into transformer layers, yielding significant influence.

| Target | ACDC[59] (test) | | | | | Cityscapes-C[48] (level-5) | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | Blur | | | | Noise | | | | Digital | | | | Weather | | | | Avg. |
| | Night | Snow | Fog | Rain | All | Motion | Defoc | Glass | Gauss | Gauss | Impul | Shot | Speck | Bright | Contr | Satur | JPEG | Snow | Spatt | Fog | Frost | |
| HGFormer | 52.7 | 68.6 | 69.9 | 72.0 | 67.2 | 64.1 | 67.2 | 61.5 | 63.6 | **27.2** | **35.7** | **32.9** | 63.1 | 79.9 | 72.9 | 78.0 | 53.6 | 55.4 | 75.8 | 75.5 | 43.2 | 59.4 |
| Ours | **70.6** | **79.5** | **76.4** | **78.2** | **77.6** | **68.5** | **71.7** | **69.7** | **68.7** | 6.2 | 23.0 | 13.1 | **63.7** | **81.5** | **78.9** | **80.6** | **68.8** | **63.8** | **73.6** | **79.5** | **47.9** | **60.0** |

Table 3. Results on **Cityscapes → ACDC (test) and Cityscapes-C (level-5)** datasets, utilizing a batch size of 8.

| Backbone | Fine-tune Method | Trainable Params* | mIoU | | | |
|---|---|---|---|---|---|---|
| | | | Citys | BDD | Map | Avg. |
| EVA02 (Large) [17, 18] | Full | 304.24M | 62.1 | 56.2 | 64.6 | 60.9 |
| | +AdvStyle [73] | 304.24M | 63.1 | 56.4 | 64.0 | 61.2 |
| | +PASTA [4] | 304.24M | 61.8 | 57.1 | 63.6 | 60.8 |
| | +GTR-LTR [53] | 304.24M | 59.8 | 57.4 | 63.2 | 60.1 |
| | Freeze | 0.00M | 56.5 | 53.6 | 58.6 | 56.2 |
| | +AdvStyle [73] | 0.00M | 51.4 | 51.6 | 56.5 | 53.2 |
| | +PASTA [4] | 0.00M | 57.8 | 52.3 | 58.5 | 56.2 |
| | +GTR-LTR [53] | 0.00M | 52.5 | 52.8 | 57.1 | 54.1 |
| | +LoRA [26] | 1.18M | 55.5 | 52.7 | 58.3 | 55.5 |
| | +AdaptFormer [6] | 3.17M | 63.7 | 59.9 | 64.2 | 62.6 |
| | +VPT [28] | 3.69M | 62.2 | 57.7 | 62.5 | 60.8 |
| | +Rein (ours) | 2.99M | **65.3** | **60.5** | **64.9** | **63.6** |
| DINOv2 (Large) [50] | Full | 304.20M | 63.7 | 57.4 | 64.2 | 61.7 |
| | +AdvStyle [73] | 304.20M | 60.8 | 58.0 | 62.5 | 60.4 |
| | +PASTA [4] | 304.20M | 62.5 | 57.2 | 64.7 | 61.5 |
| | +GTR-LTR [4] | 304.20M | 62.7 | 57.4 | 64.5 | 61.6 |
| | Freeze | 0.00M | 63.3 | 56.1 | 63.9 | 61.1 |
| | +AdvStyle [73] | 0.00M | 61.5 | 55.1 | 63.9 | 60.1 |
| | +PASTA [4] | 0.00M | 62.1 | 57.2 | 64.5 | 61.3 |
| | +GTR-LTR [4] | 0.00M | 60.2 | 57.7 | 62.2 | 60.0 |
| | +LoRA [26] | 0.79M | 65.2 | 58.3 | 64.6 | 62.7 |
| | +AdaptFormer [6] | 3.17M | 64.9 | 59.0 | 64.2 | 62.7 |
| | +VPT [28] | 3.69M | 65.2 | 59.4 | 65.5 | 63.3 |
| | +Rein (ours) | 2.99M | **66.4** | **60.4** | **66.1** | **64.3** |

Table 4. Performance Comparison of the proposed **Rein against other DGSS and PEFT methods** under the *GTAV → Cityscapes (Citys) + BDD100K (BDD) + Mapillary (Map)* generalization setting. Mark $*$ denotes trainable parameters in backbones.

The application of PEFT methods is also expanding into the field of computer vision [15, 39], with notable examples such as Visual Prompt Tuning (VPT) [28], which prepends prompts into the input sequence of transformer layers; AdaptFormer [6], replacing the MLP block in the transformer encoder with an AdaptMLP; LP-FT [39] find that fine-tuning can achieve worse accuracy than linear probing out-of-distribution; and Prompt-ICM [19], applying large-scale pre-trained models to the task of image coding for machines. Contrasting with these methods, we aim to refine feature maps for each instance within an image, thereby achieving superior performance in the realm of DGSS.

# 3. Methods

## 3.1. Preliminary

Driven by the motivation that **Leveraging Stronger pre-trained models and Fewer trainable parameters for Superior generalizability**, we choose to fine-tune VFMs with a reduced parameter set. A straightforward thought might involve a smaller decode head; however, this method merely acts as a passive receiver of feature maps from the backbone, lacking the flexibility to effectively adapt a frozen backbone for generating task-specific or scene-specific features. In contrast, we propose to embed a mechanism, named "Rein", between the layers within the backbone. Rein actively refines and forwards the feature maps from each layer to the subsequent one. This approach allows us to more effectively utilize the powerful capabilities of VFMs, much like using rein to control a horse.

| Source Domain | Cityscapes mIoU |
|---|---|
| GTAV | 66.4 |
| +Synthia | 68.1 |
| +UrbanSyn | **78.4** |
| **+1/16 of Cityscapes Training set** | **82.5** |

Table 5. **Synthetic data + 1/16 of Citys. → Citys.** val set.

Given a pre-trained VFM with parameters $\Phi_M$, consisting of a sequence of layers $L_1, L_2, \ldots, L_N$, a decode head $\mathcal{H}$ parameterized by $\theta_h$, and the Rein strategy with parameters $\theta_R$, the optimization objective can be written as:

$$\underset{\theta_R,\theta_h}{\arg\min} \sum_{i=1}^{N_d} \mathcal{L}oss(\mathcal{H}_{\theta_h}(\mathcal{F}_{\Phi_M,\theta_R}(x_i)), y_i), \tag{1}$$

where $x_i$ and $y_i$ denote the input image and its corresponding ground truth, respectively, and $N_d$ signifies the total number of samples. $\mathcal{F}_{\Phi_M,\theta_R}$ represents the forward process of VFM after applying the Rein strategy.

## 3.2. Core of Rein

For simple implementation across different VFMs, we opt not to modify MLP weights at specific positions as described in the [6, 26]. Instead, our approach focuses on refining the output feature maps at each layer within the VFMs, as illustrated in Fig. 2. Precisely, for the features $f_i$ produced by the $i$-th layer $L_i$, Rein produces enhanced feature maps for the next layer as follows:

$$\begin{aligned} f_1 &= L_1\left(Embed(x)\right) && f_1 \in \mathbb{R}^{n\times c}, \\ f_{i+1} &= L_{i+1}(f_i + \Delta f_i) && i = 1, 2, \ldots, N-1, \\ f_{out} &= f_N + \Delta f_N, \end{aligned} \tag{2}$$

where $f_i' = f_i + \Delta f_i$ symbolizes the refined feature map, $x$ is the input image, $Embed$ denotes the patch embedding layer in VFMs, $n$ represents the number of patches, $N$ denotes the number of layers, and $c$ is the dimensionality of $f_1, f_2, \ldots, f_N$. Note that the layers $L_1, L_2, \ldots, L_N$ are kept frozen, and our focus is on training an efficient module, Rein, to generate $\Delta f_i$ as follows:

$$\Delta f_i = Rein(f_i) \qquad \Delta f_i \in \mathbb{R}^{n\times c}, i = 1, 2, \ldots, N. \quad (3)$$

In the context of DGSS, an ideal $\Delta f_i$ should assist VFMs to bridge two types of gaps. The first is gap in scene between pre-training dataset and target scene, exemplified by the contrast between ImageNet [12] and urban-scene images [11, 57]. The second is task divergence between pre-training and fine-tuning, such as the differences between masked image modeling and semantic segmentation tasks.

To establish this dual bridge, Rein starts with a set of learnable tokens $T = \{T_i \in \mathbb{R}^{m\times c} \mid i \in \mathbb{N}, 1 \leq i \leq N\}$, where each token sequence $T_i$ is randomly initialized, and $m$ denotes the sequence length of $T_i$. Rein freezes the backbone and embeds knowledge learned from the fine-tuning dataset into these tokens, thereby bridging the gap in scene relative to the pre-training dataset. Moreover, considering the essential need in semantic segmentation to discern multiple instances within a single image, Rein implements an attention-inspired mechanism, which enables VFMs to make tailored adjustments to the features of distinct instances, thereby aiding VFMs in adapting to the differences between semantic segmentation and pre-training tasks. Specifically, Rein employs a dot-product operation to generate a similarity map $S_i$, which captures the associations between feature vectors in $f_i$ and the tokens in $T$:

$$S_i = f_i \times T_i^{\mathrm{T}} \qquad S_i \in \mathbb{R}^{n\times m}, \quad (4)$$

where $T_i$ represents the token sequence of the $i$-th layer, $m$ indicates the number of tokens in $T_i$. As $S$ quantitatively evaluates the relationships between various tokens and feature vectors, Rein can apply a softmax function to align each patch with a unique instance:

$$S_i = Softmax(\frac{f_i \times T_i^{\mathrm{T}}}{\sqrt{c}}). \quad (5)$$

Leveraging the feature-to-token similarity map $S_i$, we can preliminarily estimates of $\Delta f_i$ using the equation:

$$\Delta \bar{f}_i = S_i(:, 2:m) \times [\, T_i(2:m) \times W_{T_i} + b_{T_i}], \quad (6)$$

where $W_{T_i}$ and $b_{T_i}$ denote the weights and biases of a MLP, respectively. This MLP enables the transformation of $T_i$ across different feature spaces during the computation of $S_i$ and $\Delta \bar{f}_i$. Optionally, Rein can pre-calculate $T_i \times W_{T_i} + b_{T_i}$ to reduce inference time. The sum of $S_i$ equals one due to the softmax function; however, this can induce unneeded changes when all features are precise. To avoid this, $S_i(:, 2:m)$ is designed to choose columns 2 to m of $S_i$, and $T_i(2:m)$ denotes the selection of rows 2 to m of $T_i$. This strategic selection allows models to sidestep unnecessary adjustments by assigning a high value to the first token and subsequently discarding it. This approach allows the sum of each row in $S_i$ to vary from 0 to 1, thus reducing the risk of inappropriate changes.

To enhance the flexibility in feature adjustment, Rein utilizes a MLP composed of $W_{f_i}$ and $b_{f_i}$ to produce the final feature modifications $\Delta f_i$:

$$\Delta f_i = (\Delta \bar{f}_i + f_i) \times W_{f_i} + b_{f_i}. \quad (7)$$

Benefiting from these instance-level $\Delta f_i$ adjustments, Rein is capable of generating diverse modifications for various categories within a single image. The details of Rein will be explained in the next section.

### 3.3. Details of Rein

**Linking tokens to instances.** At the core of Rein, we establish an implicit yet effective linkage between tokens and instances, which has demonstrated notable performance, as detailed in Sec. 4. This connection is further reinforced by utilizing object queries, a key component in DETR[3]-style decode heads [7, 8, 71], as intermediaries. These queries are empirically proven to establish a direct association with instances. Specifically, we generate layer-wise queries $Q_i$ from our learnable tokens $T_i$ via linear transformation:

$$Q_i = T_i \times W_{Q_i} + b_{Q_i} \qquad Q_i \in \mathbb{R}^{m\times c'}, \quad (8)$$

where $W_{Q_i}$ and $b_{Q_i}$ signify the weights and biases, respectively, and $c'$ denotes the dimension of $Q_i$. However, due to the complexity arising from the large numbers of various layers in VFMs, transforming the diverse $Q_i$ into a single query $Q$ poses computational challenges. To address this, Rein computes both the maximal component $Q_{max} \in \mathbb{R}^{m\times c'}$ and the average component $Q_{avg} \in \mathbb{R}^{m\times c'}$ using the following equation:

$$\begin{aligned} Q_{max}(j,k) &= \max_{i=1,2,\ldots,N} Q_i(j,k), \\ Q_{avg}(j,k) &= \frac{1}{N}\sum_{i=1}^{N} Q_i(j,k). \end{aligned} \quad (9)$$

Subsequently, $Q$ is derived as:

$$Q = Concat([Q_{max}, Q_{avg}, Q_N]) \times W_Q + b_Q. \quad (10)$$

By mapping $T$ onto $Q$, which subsequently links to instances, Rein achieves enhanced performance with a marginal increase in parameters.

**Layer-shared MLP weights.** To address the redundancy of parameters in the layer-specific MLP weights, specifically $W_{T_i}$ in Eq. (6), $W_{f_i}$ in Eq. (7), and $W_{Q_i}$ in Eq. (8), which collectively contribute to a substantial trainable parameter count, we adopt a new strategy. Since the learnable $T_i$ is capable of producing distinct $\Delta f_i$ for each layer, we design the role of the MLP to primarily perform consistent linear transformations across different feature spaces for each layer within the backbone. To this end, we employ shared MLP weights across layers as outlined in the equations:

$$\begin{aligned}
&[W_{T_1}, b_{T_1}] = [W_{T_2}, b_{T_2}] = \ldots = [W_{T_N}, b_{T_N}], \\
&[W_{f_1}, b_{f_1}] \;= [W_{f_2}, b_{f_2}] \;= \ldots = [W_{f_N}, b_{f_N}], \\
&[W_{Q_1}, b_{Q_1}] = [W_{Q_2}, b_{Q_2}] = \ldots = [W_{Q_N}, b_{Q_N}].
\end{aligned} \tag{11}$$

**Low-rank token sequence.** Recognizing the potential for information overlap among diverse learnable tokens, such as the high similarity between tokens representing a car's headlight and a bicycle's light, Rein adopts a strategy to generate a low-rank token sequence $T$ as follows:

$$Ti = A_i \times B_i, \qquad A \in \mathbb{R}^{m\times r}, B \in \mathbb{R}^{r\times c}, \tag{12}$$

where $c$ denotes the dimension of $T_i$, $m$ is the length of sequence $T_i$, and $r$ represents the rank, with $r \ll c$. Here, matrices $A$ and $B$ are constructed as low-rank matrices. To reduce inference time, Rein can precompute and store $T$. By implementing this low-rank token sequence approach, Rein significantly reduces the number of parameter.

# 4. Experiments

## 4.1. Settings

**Visual Foundation Models.** To thoroughly assess the influence of Visual Foundation Models (VFMs) within the context of DGSS, we analyze five distinct VFMs, each with different training strategies and datasets. Our selection includes CLIP [55], a language-image pre-training model; MAE [24], known for its masked pre-training approach; SAM [38], which leverages a large-scale segmentation dataset; EVA02 [17, 18] combines CLIP with masked image modeling; and DINOv2 [50], based on self-supervised pre-training with curated dataset. For balancing precision and efficiency, we mainly employ the ViT-Large architecture for these VFMs, except SAM, which utilizes a ViT-Huge image encoder, as described in its original paper [38]. We establish two fundamental baselines for VFMs: "Full", where we fine-tune the entire network, and "Freeze", in which all backbone parameters are fixed, with training solely on the segmentation head. More details about VFMs and PEFT methods are available in the supplementary material.

**Datasets.** We evaluate VFMs and proposed methods on both real-world datasets (Cityscapes [11], BDD100K [68], Mapillary [49]) and synthetic datasets (GTAV [57], Synthia [58], UrbanSyn [22]). In detail, Cityscapes (denoted as Citys) is an autonomous driving dataset that contains 2975 training images and 500 validation images, each with the resolution of $2048 \times 1024$. BDD100K (shortened to BDD) and Mapillary (denoted by Map) offer 1,000 ($1280 \times 720$) and 2,000 ($1902 \times 1080$) validation images, respectively. GTAV, a synthetic dataset, presents 24,966 labeled images obtained from the game. Synthia, a synthetic dataset, provides 25,000 images created by photo-realistic rendering. UrbanSyn, a synthetic dataset consists of 7,539 images.

**Implementation details.** We utilize the MMSegmentation [10] codebase for our implementation. For superior performance, mask2former [8], a widely-used segmentation head, is integrated with various VFMs that serve as the backbone. Additional experiments involving other decode heads are detailed in the supplementary material. For the training phase, the AdamW optimizer [46] is employed, setting the learning rate at 1e-5 for the backbone and 1e-4 for both the decode head and the proposed Rein. Aiming to efficient training process, we utilize a configuration of 40,000 iterations with a batch size of 4, and crop images to a resolution of $512 \times 512$. Our approach includes only basic data augmentation, following Mask2Former [8]. Thanks to our streamlined training configuration and reduced number of trainable parameters, **Rein can fine-tune models like DINOv2-Large or EVA02-Large on a single RTX 3090Ti GPU within 12 hours** for superior generalization ability.

## 4.2. Comparison with State-of-The-Art Methods

In this section, we comprehensively evaluate Rein over five datasets within three generalization settings: *GTAV* $\rightarrow$ *Citys* + *BDD* + *Map*, *GTAV* + *Synthia* $\rightarrow$ *Citys* + *BDD* + *Map*, and *Citys* $\rightarrow$ *BDD* + *Map*. Rein is benchmarked against state-of-the-art (SOTA) methods, which can be classified into two groups, including domain generalized semantic segmentation (DGSS) methods[4, 9, 13, 27, 35, 40, 51, 53, 54, 64, 69, 73], and parameter-efficient fine-tuning (PEFT) approaches [6, 26, 28].

**Investigation of various VFMs.** Our analysis of VFMs and proposed Rein in the *GTAV* $\rightarrow$ *Citys* + *BDD* + *Map* setting is presented in Tables 1 and 2. In this setup, models are fine-tuned using GTAV and evaluated on Cityscapes, BDD100K, and Mapillary. Note that, due to the fixed and relatively small number of trainable parameters in the decode head (20.6M), the count of trainable parameters presented in the tables are focused solely on the backbone and the PEFT module. Our results, as detailed in Table 1, indicate that frozen VFMs significantly outperform previous DGSS methods without specialized design. Moreover, as shown in Table 2, VFMs with full parameter fine-tuning exhibit enhanced performance relative to their frozen counterparts. Remarkably, Rein achieves even superior generalization capabilities, surpassing the full parameter fine-tuning with merely an extra **1%** of trainable parameters compared to the original backbone. Visual samples for qualitative comparison are given in Fig. 3.

| Backbone | Fine-tune Method | Trainable Params* | road | side. | build. | wall | fence | pole | light | sign | vege | terr. | sky | pers. | rider | car | truck | bus | train | moto. | bicy. | mIoU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| EVA02 (Large) [17, 18] | Full | 304.24M | 89.3 | 46.9 | 89.9 | 47.7 | 45.6 | **50.1** | 56.8 | 42.2 | 88.8 | **48.4** | 89.9 | **75.8** | **49.0** | 90.5 | 45.3 | 69.2 | 55.9 | 44.4 | 55.1 | 62.2 |
| | Freeze | 0.00M | **93.1** | 52.7 | 88.0 | 47.4 | 31.1 | 41.7 | 46.0 | 39.6 | 85.7 | 41.4 | 89.5 | 67.5 | 39.7 | 89.0 | 47.0 | 72.8 | 46.3 | 19.2 | 35.2 | 56.5 |
| | Rein-core | 52.84M | 91.1 | 53.8 | 90.0 | 50.3 | 47.7 | 46.6 | 56.4 | 42.9 | 87.8 | 44.2 | 90.4 | 73.5 | 44.2 | **91.8** | **58.1** | 77.2 | 57.3 | 43.4 | 57.3 | 63.4 |
| | + Rein-link | 59.33M | 90.9 | 48.5 | 90.0 | 52.6 | **49.4** | 49.1 | **57.2** | 39.8 | 88.9 | 46.5 | **90.5** | 74.4 | 44.0 | 91.0 | 52.3 | 80.7 | 67.3 | 44.3 | **60.3** | 64.1 |
| | + Rein-share | 5.02M | 92.7 | **54.3** | 90.0 | 51.8 | 48.6 | 48.8 | 55.3 | **45.0** | 88.9 | 46.7 | 89.8 | 73.7 | 43.3 | 90.6 | 49.5 | 81.1 | 69.6 | 41.7 | 50.2 | 63.4 |
| | + Rein-lora | 2.99M | 91.7 | 51.8 | **90.1** | **52.8** | 48.4 | 48.2 | 56.0 | 42.0 | **89.1** | 44.1 | 90.2 | 74.2 | 47.0 | 91.1 | 54.5 | **84.1** | **78.9** | **47.2** | 59.4 | **65.3** |
| DINOv2 (Large) [50] | Full | 304.20M | 89.0 | 44.5 | 89.6 | 51.1 | 46.4 | 49.2 | **60.0** | 38.9 | 89.1 | 47.5 | **91.7** | 75.8 | **48.2** | 91.7 | 52.5 | **82.9** | **81.0** | 30.4 | 49.9 | 63.7 |
| | Freeze | 0.00M | 92.1 | 55.2 | 90.2 | 57.2 | 48.5 | 49.5 | 56.7 | 47.7 | 89.3 | 47.8 | 91.1 | 74.2 | 46.7 | 92.2 | 62.6 | 77.5 | 47.7 | 29.6 | 47.2 | 61.1 |
| | Rein-core | 52.84M | 92.4 | 57.8 | 90.6 | 56.8 | 50.7 | 50.5 | 57.5 | 44.8 | **89.8** | 47.0 | 91.1 | **75.9** | 47.2 | 91.9 | 60.1 | 80.3 | 59.8 | 37.9 | 52.3 | 64.9 |
| | + Rein-link | 59.33M | 91.2 | 55.5 | 90.6 | 55.6 | 52.5 | 51.1 | 59.7 | 45.1 | **89.8** | 47.1 | 91.1 | 75.8 | 47.1 | **92.6** | **64.6** | 82.2 | 65.5 | **40.4** | 52.7 | 65.8 |
| | + Rein-share | 5.02M | **93.5** | **61.2** | **90.7** | 57.7 | 53.2 | **52.4** | 58.0 | **50.1** | 89.7 | **49.9** | 90.7 | 74.8 | 45.0 | 91.7 | 58.5 | 80.1 | 66.3 | 36.9 | 50.7 | 65.8 |
| | + Rein-lora | 2.99M | 92.4 | 59.1 | **90.7** | **58.3** | **53.7** | 51.8 | 58.2 | 46.4 | **89.8** | 49.4 | 90.8 | 73.9 | 43.3 | 92.3 | 64.3 | 81.6 | 70.9 | **40.4** | **54.0** | **66.4** |

Table 6. Ablation Study about Rein under *Cityscapes* → *BDD100K* generalization in terms of mIoU. Components are sequentially incorporated. To better illustrate the gains contributed by each component, we employ varying shades of yellow to demonstrate the relative performance of the Freeze and Rein methods. The best results across all methods are **highlighted**.

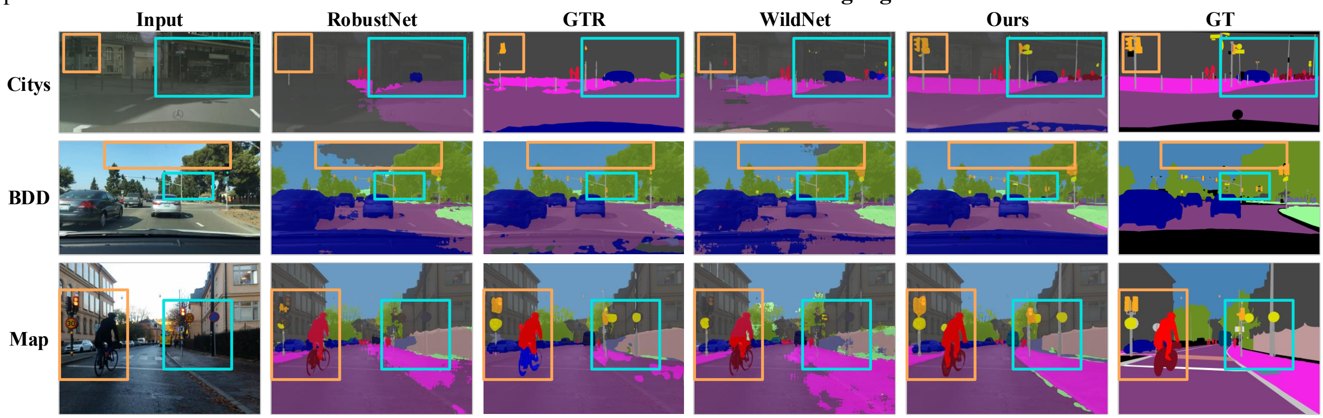



Figure 3. Qualitative Comparison under ***GTAV*** → ***Cityscapes (Citys)*** **+** ***BDD100K (BDD)*** **+** ***Mapillary (Map)*** **generalization setting.**

**Comparing Rein with SOTA.** We conduct a comprehensive performance comparison of the proposed **Rein against existing DGSS and PEFT** methods under the *GTAV* → *Citys* + *BDD* + *Map* setting, as detailed in Table 4. Owing to the robust feature extraction capabilities inherent in VFMs, DGSS methods, which typically enhance generalizability through strong data augmentation or consistency constraints, (*e.g*., AdvStyle, PASTA, and GTR), do not exhibit significant performance improvement. On the other hand, PEFT methods have demonstrated notable advancements. For instance, AdaptFormer outperforms the "Freeze" baseline using EVA02 as the backbone, while VPT shows improved performance over "Full" with DINOv2. Employing the same backbones (DINOv2 and EVA02), proposed Rein achieves superior performance and surpass previous DGSS and PEFT methods.

**Real-to-Real generalization of Rein.** The generalization from one real-world dataset to others is pivotal for practical applications in the field. To this end, we conduct experiments under the ***Citys*** → ***ACDC***, *Citys* → *Cityscapes-C*, and *Citys* → *BDD* + *Map* generalization setting. As shown in Table 3 and 7, Rein, when coupled with the DINOv2-Large, demonstrates superior performance across all datasets. This underscores the effectiveness of Rein in generalizing to diverse real-world scenarios.

**Synthetic-to-real generalization of Rein.** As Tab. 5 illustrates, trained on **synthetic** *UrbanSyn+GTAV+Synthia* datasets, Rein achieved a **78.4% mIoU** on the Cityscapes validation set. Further improvement is possible with additional synthetic data and higher-quality images generated by diffusion models, like [1]. This result can also be a valuable pre-trained weight for data-efficient training, **reaching an** 82.5% **mIoU with 1/16 of Cityscapes training set**. This is a significant performance for semi-supervised semantic segmentation.

**More backbones.** We extend our analysis to integrating Rein with Convolutional Networks, such as ResNet and ConvNeXt, and smaller scale architectures like DINOv2-S/B. As shown in Table 8, our findings reveal that Rein exhibits remarkable performance with diverse backbones.

### 4.3. Ablation Studies and Analysis

We conduct extensive ablation studies within two settings: *GTAV* → *Citys* and *GTAV* → *Citys* + *BDD* + *Map*.

**Analysis of the key components.** Table 6 is dedicated to thoroughly examining the effectiveness of each component within Rein. In the *GTAV* → *Citys* generalization setting, we sequentially incorporate different components of Rein and assess their impact. Interestingly, we observe that the "Freeze" occasionally exhibit better recognition for specific

| Methods | Backbone | Trainable Parameters* | mIoU | | |
|---|---|---|---|---|---|
| | | | BDD | Map | Avg. |
| IBN [51] | ResNet50 [23] | 23.58M | 48.6 | 57.0 | 52.8 |
| DRPC [69] | ResNet50 [23] | 23.58M | 49.9 | 56.3 | 53.1 |
| GTR [53] | ResNet50 [23] | 23.58M | 50.8 | 57.2 | 54.0 |
| SAN-SAW [54] | ResNet50 [23] | 23.58M | 53.0 | 59.8 | 56.4 |
| WildNet [40] | ResNet101 [23] | 42.62M | 50.9 | 58.8 | 54.9 |
| HGFormer [13] | Swin-L [45] | 196.03M | 61.5 | 72.1 | 66.8 |
| Freeze | EVA02-L [17] | 0.00M | 57.8 | 63.8 | 60.8 |
| Rein (Ours) | EVA02-L [17] | 2.99M | 64.1 | 69.5 | 66.8 |
| Freeze | DINOv2-L [50] | 0.00M | 63.4 | 69.7 | 66.7 |
| Rein (Ours) | DINOv2-L [50] | 2.99M | **65.0** | **72.3** | **68.7** |

Table 7. Performance Comparison of the **Rein against other DGSS methods** under *Cityscapes → BDD100K (BDD) +Mapillary (Map)* generalization. The best results are **highlighted**.

| Avg. mIoU | ResNet (50) | ResNet (101) | ConvNeXt (Large) | DINOv2 (S) | DINOv2 (B) |
|---|---|---|---|---|---|
| Full | 38.9 | 46.1 | 52.2 | 51.8 | 56.7 |
| Ours | **46.6** | **46.3** | **55.5** | **55.7** | **59.1** |

Table 8. Results for **ConvNets and smaller backbones**.

categories, *e.g*., 'road, sidewalk', compared to the "Full". This suggests that VFMs lose some pre-training knowledge during fine-tuning, and "Freeze" helps to prevent. Similarly, our methods mitigate this knowledge forgetting. Furthermore, our methods show improved recognition capabilities for the majority of the 19 categories. For example, in recognizing 'wall, motorcycle, bicycle', our approach significantly outperforms both the "Full" and "Freeze" baselines.

Overall, "Rein-core" boosts the average performance across 19 classes. Furthermore, "Rein-link" further boosts accuracy for certain objects, including 'car, bus, train, motorcycle', especially for DINOv2. Employing layer-shared MLP weights and low-rank token sequence efficiently reduces the number of trainable parameters and positively influences the performance of the model.

**Study on token length $m$.** The core component of Rein is learnable tokens $T \in \mathbb{R}^{m\times c}$. We explored various lengths $m$ for the token sequence, ranging from 25 to 200. As demonstrated in Fig. 4, models with $m = 100$ and $m = 150$ both achieve a strong mIoU of 64.3%. We ultimately selected $m = 100$ as the most suitable parameter.

**Study on rank $r$.** As shown in Table 9, we turn attention to the effect of rank $r$ on model performance. With DINOv2 as the backbone, the optimal results are observed at $r = 16$ and $r = 32$. Consequently, unlike LoRA [26], we opt for a comparatively higher value of $r = 16$ for our model.

**Speed, memory, and storage.** For practical applications, training speed, GPU memory usage, and model storage requirements are crucial. As shown in Table 10, compared to "Full" baseline, proposed Rein improves training speed and reduces GPU memory usage. A significant advantage of Rein is that models trained under different settings can share the same backbone parameters. This means that for switch

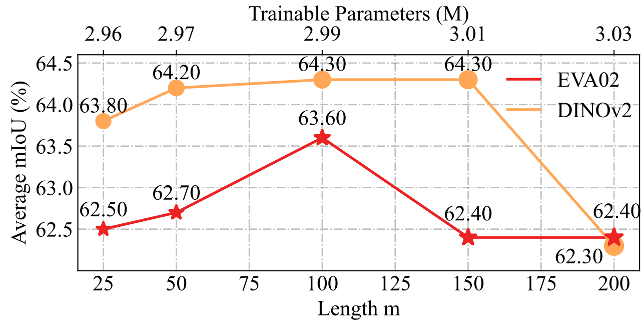



Figure 4. Ablation study on token length $m$.

| Rank $r$ | | 4 | 8 | 16 | 32 | 64 |
|---|---|---|---|---|---|---|
| Params | | 2.67M | 2.77M | 2.99M | 3.42M | 4.28M |
| DINOv2 (Large) [50] | Citys | 65.8 | 66.1 | 66.4 | 66.1 | 66.4 |
| | BDD | 60.2 | 60.3 | 60.4 | 60.7 | 61.0 |
| | Map | 65.2 | 65.1 | 66.1 | 65.9 | 65.0 |
| | Avg. | 63.7 | 63.9 | **64.3** | **64.3** | 64.1 |

Table 9. Ablation study on lora dim $r$.

| VFMs | Method | Training Time | GPU Memory | Storage |
|---|---|---|---|---|
| DINOv2 (Large) | Full | 11.2 h | 14.7 GB | 1.22 GB |
| | Rein | 9.5 h | 10.0 GB | 1.23 GB |

Table 10. Training Time, GPU Memory, and Storage.

in diverse tasks and settings, we can only store and swap the rein weights (0.01GB) and head weights (0.08GB), rather than all parameters.

## 5. Conclusions

In this paper, we assess and harness Vision Foundation Models (VFMs) in the context of DGSS. Driven by the motivation that **Leveraging Stronger pre-trained models and Fewer trainable parameters for Superior generalizability**, we first investigate the performance of VFMs under diverse DGSS settings. Subsequently, we introduce a robust fine-tuning approach, namely **Rein**, to parameter-efficiently harness VFMs for DGSS. With a fewer trainable parameters, Rein significantly enhance generalizability of VFMs, outperforming SOTA methods by a large margin. Rein can be seamlessly integrated as a plug-and-play adapter for existing VFMs, improving generalization with efficient training. Extensive experiments demonstrate the substantial potential of VFMs in the DGSS field, validating the effectiveness of proposed Rein in harnessing VFMs for DGSS.

## 6. Acknowledgements

This work was supported in part by the Anhui Provincial Key Research and Development Plan 202304a05020072, in part by the Postdoctoral Fellowship Program of CPSFGZB20230713, and in part by the National Natural Science Foundation of China under Grant 61727809.

## References


[1] Yasser Benigmim, Subhankar Roy, Slim Essid, Vicky Kalogeiton, and Stéphane Lathuilière. Collaborating foundation models for domain generalized semantic segmentation, 2023. 7

[2] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models, 2022. 3

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 5

[4] Prithvijit Chattopadhyay, Kartik Sarangmath, Vivek Vijaykumar, and Judy Hoffman. Pasta: Proportional amplitude spectrum training augmentation for syn-to-real domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19288–19300, 2023. 1, 2, 3, 4, 6

[5] Lin Chen, Zhixiang Wei, Xin Jin, Huaian Chen, Miao Zheng, Kai Chen, and Yi Jin. Deliberated domain bridging for domain adaptive semantic segmentation. In *Advances in Neural Information Processing Systems*, pages 15105–15118. Curran Associates, Inc., 2022. 2

[6] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022. 2, 4, 6

[7] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021. 5

[8] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 2, 5, 6

[9] Sungha Choi, Sanghun Jung, Huiwon Yun, Joanne T Kim, Seungryong Kim, and Jaegul Choo. Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11580–11590, 2021. 1, 3, 6

[10] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 6

[11] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 1, 2, 3, 5, 6

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2, 5

[13] Jian Ding, Nan Xue, Gui-Song Xia, Bernt Schiele, and Dengxin Dai. Hgformer: Hierarchical grouping transformer for domain generalized semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15413–15423, 2023. 2, 6, 8

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[15] Mohammad Fahes, Tuan-Hung Vu, Andrei Bursuc, Patrick Pérez, and Raoul de Charette. A simple recipe for language-guided domain generalized segmentation. *arXiv preprint arXiv:2311.17922*, 2023. 4

[16] Qi Fan, Mattia Segu, Yu-Wing Tai, Fisher Yu, Chi-Keung Tang, Bernt Schiele, and Dengxin Dai. Towards robust object detection invariant to real-world domain shifts. In *The Eleventh International Conference on Learning Representations (ICLR 2023)*. OpenReview, 2023. 3

[17] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *arXiv preprint arXiv:2303.11331*, 2023. 1, 2, 3, 4, 6, 7, 8

[18] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue

Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19358–19369, 2023. 1, 3, 4, 6, 7
[19] Ruoyu Feng, Jinming Liu, Xin Jin, Xiaohan Pan, Heming Sun, and Zhibo Chen. Prompt-icm: A unified framework towards image coding for machines with task-driven prompts. *arXiv preprint arXiv:2305.02578*, 2023. 4
[20] Ziyang Gong, Fuhao Li, Yupeng Deng, Wenjun Shen, Xianzheng Ma, Zhenming Ji, and Nan Xia. Train one, generalize to all: Generalizable semantic segmentation from single-scene to all adverse scenes. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 2275–2284, 2023. 2
[21] Ziyang Gong, Fuhao Li, Yupeng Deng, Deblina Bhattacharjee, Xiangwei Zhu, and Zhenming Ji. Coda: Instructive chain-of-domain adaptation with severity-aware visual prompt tuning, 2024. 2
[22] Jose L. Gómez, Manuel Silva, Antonio Seoane, Agnès Borrás, Mario Noriega, Germán Ros, Jose A. Iglesias-Guitian, and Antonio M. López. All for one, and one for all: Urbansyn dataset, the third musketeer of synthetic driving scenes, 2023. 6
[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 2, 3, 8
[24] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 1, 2, 3, 6
[25] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019. 2, 3
[26] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 2, 3, 4, 6, 8
[27] Wei Huang, Chang Chen, Yong Li, Jiacheng Li, Cheng Li, Fenglong Song, Youliang Yan, and Zhiwei Xiong. Style projected clustering for domain generalized semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3061–3071, 2023. 2, 6
[28] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022. 2, 4, 6
[29] Xueying Jiang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Domain generalization via balancing training difficulty and model capability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18993–19003, 2023. 2
[30] Mengmeng Jing, Xiantong Zhen, Jingjing Li, and Cees GM Snoek. Order-preserving consistency regularization for domain adaptation and generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18916–18927, 2023.
[31] Christoph Kamann and Carsten Rother. Increasing the robustness of semantic segmentation models with painting-by-numbers. In *European Conference on Computer Vision*, pages 369–387. Springer, 2020. 2
[32] Juwon Kang, Sohyun Lee, Namyup Kim, and Suha Kwak. Style neophile: Constantly seeking novel styles for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7130–7140, 2022. 1
[33] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 2
[34] Hyeonseong Kim, Yoonsu Kang, Changgyoon Oh, and Kuk-Jin Yoon. Single domain generalization for lidar semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17587–17598, 2023. 1
[35] Jin Kim, Jiyoung Lee, Jungin Park, Dongbo Min, and Kwanghoon Sohn. Pin the memory: Learning to generalize semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4350–4360, 2022. 2, 6
[36] Namyup Kim, Taeyoung Son, Jaehyun Pahk, Cuiling Lan, Wenjun Zeng, and Suha Kwak. Wedge: web-image assisted domain generalization for semantic segmentation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9281–9288. IEEE, 2023. 1
[37] Sunghwan Kim, Dae-hwan Kim, and Hoseong Kim. Texture learning domain randomization for domain generalized segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 677–687, 2023. 2
[38] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 1, 2, 3, 6
[39] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022. 4
[40] Suhyeon Lee, Hongje Seong, Seongwon Lee, and Euntai Kim. Wildnet: Learning domain generalized semantic segmentation from the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9936–9946, 2022. 2, 3, 6, 8
[41] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 2, 3
[42] Fuhao Li, Ziyang Gong, Yupeng Deng, Xianzheng Ma, Renrui Zhang, Zhenming Ji, Xiangwei Zhu, and Hong Zhang.

Parsing all adverse scenes: Severity-aware semantic segmentation with mask-enhanced cross-domain consistency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13483–13491, 2024. 2

[43] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021. 2

[44] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68, 2022. 2

[45] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3, 8

[46] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[47] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, pages 116–131, 2018. 3

[48] Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S Ecker, Matthias Bethge, and Wieland Brendel. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, 2019. 4

[49] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE international conference on computer vision*, pages 4990–4999, 2017. 6

[50] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1, 2, 3, 4, 6, 7, 8

[51] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 464–479, 2018. 1, 3, 6, 8

[52] Xingang Pan, Xiaohang Zhan, Jianping Shi, Xiaoou Tang, and Ping Luo. Switchable whitening for deep representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1863–1871, 2019. 3

[53] Duo Peng, Yinjie Lei, Lingqiao Liu, Pingping Zhang, and Jun Liu. Global and local texture randomization for synthetic-to-real semantic segmentation. *IEEE Transactions on Image Processing*, 30:6594–6608, 2021. 1, 2, 3, 4, 6, 8

[54] Duo Peng, Yinjie Lei, Munawar Hayat, Yulan Guo, and Wen Li. Semantic-aware domain generalized segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2594–2605, 2022. 2, 6, 8

[55] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 2, 3, 6

[56] Nikhil Reddy, Abhinav Singhal, Abhishek Kumar, Mahsa Baktashmotlagh, and Chetan Arora. Master of all: Simultaneous generalization of urban-scene segmentation to all adverse weather conditions. In *European Conference on Computer Vision*, pages 51–69. Springer, 2022. 1

[57] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 102–118. Springer, 2016. 1, 2, 3, 5, 6

[58] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[59] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10765–10775, 2021. 4

[60] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 1, 2, 3

[61] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1, 3

[62] Zhiqiang Tang, Yunhe Gao, Yi Zhu, Zhi Zhang, Mu Li, and Dimitris N. Metaxas. Crossnorm and selfnorm for generalization under distribution shifts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 52–61, 2021. 1, 2

[63] Jan-Aike Termöhlen, Timo Bartels, and Tim Fingscheidt. A re-parameterized vision transformer (revt) for domain-generalized semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4376–4385, 2023. 1

[64] Wei Wang, Zhun Zhong, Weijie Wang, Xi Chen, Charles Ling, Boyu Wang, and Nicu Sebe. Dynamically instance-guided adaptation: A backward-free approach for test-time domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24090–24099, 2023. 6

[65] Zhixiang Wei, Lin Chen, Tao Tu, Pengyang Ling, Huaian Chen, and Yi Jin. Disentangle then parse: Night-time semantic segmentation with illumination disentanglement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 21593–21603, 2023. 2

[66] Zhenyao Wu, Xinyi Wu, Xiaoping Zhang, Lili Ju, and Song Wang. Siamdoge: Domain generalizable semantic segmentation using siamese network. In *European Conference on Computer Vision*, pages 603–620. Springer, 2022. 1

[67] Qi Xu, Liang Yao, Zhengkai Jiang, Guannan Jiang, Wenqing Chu, Wenhui Han, Wei Zhang, Chengjie Wang, and Ying Tai. Dirl: Domain-invariant representation learning for generalizable semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2884–2892, 2022. 1, 2

[68] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020. 6

[69] Xiangyu Yue, Yang Zhang, Sicheng Zhao, Alberto Sangiovanni-Vincentelli, Kurt Keutzer, and Boqing Gong. Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2100–2110, 2019. 6, 8

[70] Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*, 2021. 2, 3

[71] Bowen Zhang, Zhi Tian, Quan Tang, Xiangxiang Chu, Xiaolin Wei, Chunhua Shen, et al. Segvit: Semantic segmentation with plain vision transformers. *Advances in Neural Information Processing Systems*, 35:4971–4982, 2022. 5

[72] Yuhang Zhang, Shishun Tian, Muxin Liao, Guoguang Hua, Wenbin Zou, and Chen Xu. Learning shape-invariant representation for generalizable semantic segmentation. *IEEE Transactions on Image Processing*, 2023. 1

[73] Zhun Zhong, Yuyang Zhao, Gim Hee Lee, and Nicu Sebe. Adversarial style augmentation for domain generalized urban-scene segmentation. *Advances in Neural Information Processing Systems*, 35:338–350, 2022. 1, 2, 3, 4, 6

[74] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 2