Towards Domain Generalized Segmentation with Transformer

Qi Yan* qi.yan@ece.ubc.ca
Zhongze Chen* chenzze1@student.ubc.ca
Menghong Huang* ritalian@student.ubc.ca
Rui Yao* rruiyao@student.ubc.ca

Abstract

Semantic segmentation has progressed substantially in the recent decade, thanks to the emergence of expressive deep learning models. However, domain shift still poses severe challenges to existing methods in that the neural models tend to deliver degraded performance when faced with data out of the training set distribution. Our work is dedicated to improving domain generalization for segmentation tasks. Based on the state-of-the-art visual transformer backbone, we incorporate a self-supervised framework of masked image modeling in the pre-training phase. Further, we propose patch-wise style-mixing with pixel-wise contrastive learning in the fine-tuning stage. We evaluate our proposed approaches on the ACDC dataset [17], which contains four adverse and challenging visual conditions of drive scenes. Our method achieves over 3% mIoU increase on four domains compared with the state-of-the-art CNN-based algorithm and about 1% to 2% growth over the vanilla transformer-based algorithm. We hope the proposed method may offer new perspectives on domain generalization for complex real-world applications. Our code is available at: https://github.com/czz1997/Swin-Transformer-Semantic-Segmentation.

1 Introduction

(a) Normal condition. (b) Adverse condition.

Figure 1: **(a)** Image semantic segmentation in normal condition [11]. **(b)** Domain shift: Semantic segmentation in snowy scene with performance greatly deteriorated [17].

Image segmentation tasks have been long-standing critical topics in the computer vision community. At the core of the task is the problem of grouping pixels by some criteria of interest. Commonly-adopted segmentation tasks include panoptic segmentation, instance segmentation and semantic segmentation which has dense pixel-level prediction. However, segmentation applications still suffer unexpected performance drop due to the diverse and unseen scenarios they face in the deployment of the real world. This problem is called domain shift where the test data is out of distribution of source domains used in training. The trained model can be specific to the source domains and lack of the ability to well generalize to arbitrary different domains in the wild. Fig.1 depicts the model well-implemented in driving scene in clear weather can have severe degraded semantic segmentation on the snowy scene. Sakaridis et. al [17] shows that even HRNet, one of the recent state-of-the-art

*All authors contribute equally.

segmentation networks, has about 5.4% of mIoU drop when it is deployed on adverse weather conditions compared with that in the daytime and clear environment.

It is crucial to improve the domain generalization in the industry such as autonomous driving as it requires the satisfactory performance of semantic segmentation even in adverse conditions of driving scenes. To gain consistent performance on unseen or unknown domains, domain generalization (DG) techniques are studied to learn domain-invariant feature representations without the auxiliary samples from target domains.Most of DG approaches are targeted to classification task while DG on segmentation is less studied [4]. Through front-to-end architecture design, there are potential approaches to improve the DG of segmentation. For backbone design, pioneered by vision transformer(ViT) [6], transformer based backbone recently shows strong representation capabilities that foster more potential improvement on multiple vision tasks. In the model pretraining phase, self-supervised learning (SSL) prompts the model to generalize in an excellent way and provide high-quality representations of the inputs,which then transfer well to downstream tasks and achieve outperformed performance than supervised counterpart [9, 8]. There are multiple DG techniques performed in fine tuning phase. Data augmentation [23, 13] is widely studied, which use various transformations to increase the diversity of source domains in training. Contrastive representation learning in supervised setting[10, 19] also shows good regularization on learned representations so that enhance the robustness of vision tasks like semantic segmentation.

In this paper, we aim to study the segmentation architecture that is robust against data distribution shift across domains and generalizes well to any unseen domains. Our work explores leveraging SSL, data augmentation techniques and the contrastive representation learning in transformer architecture to improve domain generalization on segmentation task. The main contributions of this paper are summarized as follows:

- We explore SSL pretraining with masked image modeling in Swin-Transformer backbone to learn generalized representation and benefit the domain generalized segmentation.
- We propose patch-wise style mixing based on MixStyle to enhance source domain diversity so that our model can learn from more novel domains during training.
- We present contrastive learning with style mixing to facilitate learning domain-invariant features.
- Our segmentor achieve considerable improvements over baseline models on the challenging adverse domain dataset, ACDC. The results illustrate that style mixing with pixel-wise contrastive learning can boost the segmentation performance and better generalize to unseen domains.

2 Related Work

Transformer backbone. While the performance of vision applications with CNN backbones is increasingly saturating, recent studies show transformer based backbones is competitive in representation learning and have substantial room for improvement in dense recognition tasks such as segmentation [7]. Vision Transformer(ViT) [6] is the first to use a pure transformer encoder as backbone to process image patches for classification and shows better accuracy and computational efficiency than CNN-based classification models. Swin Transformer [15](SwinTF) introduces useful inductive biases of locality by processing self-attention within windows and re-partitioning windows in each layer to obtain cross-window connections. It merges image patches after each transformer block except the last one to obtain the hierarchical feature maps. This approach derives a transformer with linear computational complexity to input image size and compatible for various image scale and high-resolution images. The second version of SwinTF [14] already upgrades the capacity and resolution to 1536 x 1536. It can work as a general backbone for multiple CV tasks and its informative hierarchical feature maps benefit to following segmentation tasks.

Domain generalization. Self-supervised learning(SSL) recently also shows revolutionary trend in computer vision that makes pre-trained representations well-generalized to various downstream tasks [21]. There are two main frameworks for SSL pretraining in literature: masked signal modeling framework and contrastive learning paradigms. Simmin [22] and MAE [8] are both based on masked image modeling. By learning to predict the invisible part that are masked,the resulting pre-trained transformers obtain strong representations. Contrastive learning framework learns similar/dissimilar

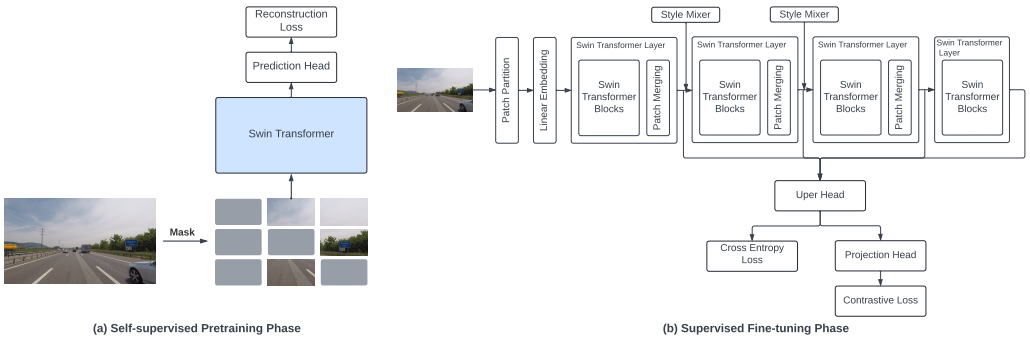

Figure 2: Overview of our proposed approach. **(a)** In the pretraining phase, we adopt the masked signal modeling to formulate a powerful SSL task. The network is trained to reconstruct the missing portions of the input image. **(a)** On top of that, we design patch-wise style mixer to synthesize novel domains and adopt pixel-wise contrastive learning to learn domain-invariant pixel embedding.

representations from data that are organized into similar/dissimilar pairs and uses the contrastive loss like InfoNCE [16] for training. Chen et. al [2] explores the recipe of pre-training self-supervised ViT in contrastive learning framework and introduces the MoCo-V3 to improve stability. MoBY [21] is based on momentum contrast framework to implement SSL pretraining on SwinTF backbone and achieve top performance on image classification.

Data augmentation is one of effective domain generalization techniques that regularize the model by increasing the diversity of source domains. Zhou et al.[23] proposed MixStyle that mix the feature statistics in instance level at the bottom layers of CNN to synthesize new style thereby increase the diversity of the source domains. Li et al. [13] synthesize feature statistics to simulate uncertain domain shift by randomly sample variants from the estimated Gaussian distribution.

Supervised contrastive learning is also widely used in fine-tuning phase to boost model generalization ability. As indicated in [10], the contrastive learning loss could also be used in the supervised learning setting. Closely related to our project, Wang et al. [19] provide a pixel-wise contrastive learning with the CNN backbone,termed ContrastiveSeg.This method exploits the structures of labeled data and the relations across images by gathering pixel embeddings with same class closer and diverge them from different classes. It eventually enhances the robustness of semantic segmentation.

As current success of domain generalization methods like Mixstyle is mainly on image classification, our project extend the idea to segmentation task with much dense prediction. Self-supervised learning and supervised contrastive learning also show potentials to enhance model generalization ability while they are less studied on domain generalization. Thurs we also explore their implementation on transformer embedding space for domain generalized segmentation.

3 Method

Our work is based on the Swin Transformer backbone [15]. The overall structure of the proposed method is shown in Fig. 2. In the self-supervised learning stage, we deploy masked signal modeling for generalized representation learning. In supervised learning stage, we develop patch-wise style mixing with contrastive loss for domain-invariant feature embedding learning.

3.1 Masked Signal Modeling SSL

SSL for vision tasks has been predominated by the contrastive learning in recent works [1, 9], with the pretext task being a classification problem at the latent space. In the language domain, the prevailing option, however, is to predict a portion of input signals that are masked out, *e.g.,* sentence auto-completion. The idea of masked signal modeling as such formulates a powerful SSL pretext task requiring the model to encode sufficient information about the data distribution.

The visual transformer backbone employs the patch-wise visual representations as its token, making it convenient to implement the patch-aligned imagery masking and prediction task as seen in Fig. 2.

Figure 3: Reference batch generated by shuffling patches across instance. x_1 and x_2 are instances of input batch and y_1 and y_2 are instances of reference batch. $x_{i,j}$ is the j_{th} patch of the i_{th} instance of the original batch. Different colors represent different domains.

The idea is intuitive and straightforward: we mask out several input image patches and adopt a light prediction head to reconstruct the missing components. During the embedding stage, we use a learnable mask token for the missing patches to be aligned with the standard practice in the NLP.

The visual masked signal modeling is equivalent to the image inpainting and is a generic SSL method applicable to both CNN and transformer backbones. While it has been proven helpful for visual transformers [3, 22] recently, whether it could still be promising for domain generalization tasks remain unknown. The nuance is that the learned representations encoding inpainting information may not be necessarily favorable for generalizing to new visual domains.

3.2 Patch-wise Style Mixing

In supervised fine-tuning phase, to make our model more robust to domain shift problems, we first make our model learn from more diverse domains by generating synthetic domains that are not in the training set. We design the patch-wise style mixer for Transformer based on MixStyle [23] to diversify training domains.

In MixStyle, it generates a reference batch by randomly shuffling the instances in the original batch and then mix the feature statistics of the instance in the input batch and the corresponding instance in the reference batch with an instance-wise weight λ. With mixed feature statistics, it uses AdaIN to apply styles to the feature map and then the model can learn from unseen domains. Since MixStyle mixes styles of different instances and also adopts instance-wise weights, it is an instance-level style mixing and is therefore heavily influenced by the batch size. When batch size grows, the extra instances in the batch can help synthesize much more diverse domains for the model to learn. However, large batch size is not awalys an option. For instance, in semantic segmentation with Swin Transformer, a GPU with 16G VRAM is only able to accommodate 2 instances per batch. With limited instances, it is unlikely that instance-level style mixing will be able to produce diverse domains and as a result, MixStyle is prone to be ineffective.

To address the batch size issue with MixStyle, we propose to perform patch-wise style mixing instead of mixing styles in the instanc level. The idea behind patch-wise style mixing is that a batch may have limited number of instances, whereas each instance will have considerable number of patches since we are using transformer as the backbone. Each patch can have different feature statistics than other patches of the same image, because of different illumination, angles, etc. When we mix styles of different patches, we can synthesize much more diverse styles even with very limited batch size.

Specifically, given an input batch x, we first shuffle patches across image to generate a reference batch y, as shown in Fig. 3. Then we compute the mixed feature statistics $\gamma_{i,j}$ and $\beta_{i,j}$ for each patch by

$$\gamma_{i,j} = \lambda_{i,j}\sigma(x_{i,j}) + (1 - \lambda_{i,j})\sigma(y_{i,j}) \tag{1}$$

$$\beta_{i,j} = \lambda_{i,j}\mu(x_{i,j}) + (1 - \lambda_{i,j})\mu(y_{i,j}) \tag{2}$$

where $\lambda_{i,j}$ is sampled patch-wise from Beta distribution. We use patch-wise weights instead of instance-wise weights to make the synthesized domains more diverse. Finally, we apply synthesized styles to each patch by replacing the learned variable β and γ in layer norm with computed mixed

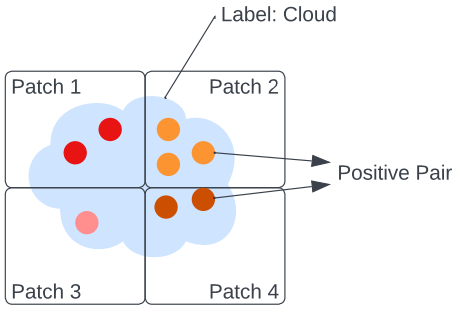

Figure 4: Example of contrastive learning with style mixing. The figure shows a region of an image where pixels are labeled as cloud. The region spreads to four patches, and pixels (circles) of different patches are in different domains (colors) because of patch-wise style mixing. Useful postive pairs can be constructed by sampling pixels from different patches.

feature statistics,

$$x_{i,j(mix)} = \frac{x_{i,j} - \mu(x_{i,j})}{\sigma(x_{i,j})} * \gamma_{i,j} + \beta_{i,j} \tag{3}$$

Follow MixStyle, we apply our patch-wise style mixer after the first two layers, which are transformer's equivalent of res1-2. It replaces the layer norm after the patch merging and before the next transformer block.

3.3 Contrastive Learning with Style Mixing

In addition to synthesize diverse domains for training, we also make the feature embeddings invariant to domain changes by explicitly pulling feature embeddings of the same class but different domains together. We propose contrastive learning with style mixing, as shown in Fig. 4, which combines patch-wise style mixing and pixel-wise contrastive learning [19].

In pixel-wise contrastive learning, pixel embeddings with the same class label will be considered as the positive pairs and pixel embeddings with different class labels will be treated as the negative pairs. Similar to MixStyle, this method is also heavily limited by the batch size and requires memory bank to be effective, as it aims for cross image contrastive learning.

However, since we are using transformer as the backbone, pixels from the same object in an image can spread across multiple patches, and each patch may be in very different domains than its neighboring patches because of the style mixing. As a result, by pulling positive pairs of pixel embeddings from the same object in an image but from different patches together, the model can learn to make pixel embeddings invariant to domain shift. In this way, even if the batch size is small, we can still make pixel-wise contrastive learning effective.

Specifically, follow [19], we take the output before the pixel classification layer and project the output to pixel embeddings. After L2 normalizing the embeddings, we perform hard example mining and use the hard examples to contruct postive and negative pairs to compute contrastive loss.

4 Experiments

4.1 Dataset

The Adverse Conditions Dataset with Correspondences (ACDC) [17] is used for training and testing our semantic segmentation models on adverse visual conditions. It comprises a large set of 4006 images evenly distributed across fog, nighttime, rain, and snow conditions. Each image comes with a high-quality refined pixel-level semantic annotation.

We adopt the leave-one-domain-out strategy [12] to preprocess the overall dataset and intentionally formulate a challenging domain generalization task. Initially, each domain has a training set, a validation set, and a hidden testing set. To speed up our validation, we discard all the testing set in

Method	Backbone	SSL	Fog	Night	Rain	Snow	Comment
HRNet [18]	N/A	N/A	71.9	30.9	69.2	66.3	Baseline
SwinTF vanilla [15]	Small	None	75.2	40.5	70.5	67.6	
SwinTF+**Style Mixing+Contrast**			**76.1**	**42.2**	**72.2**	**70.0**	Ours
SwinTF+**SSL+Style Mixing+Contrast**		SimMIM [22]	38.4	9.67	40.4	35.7	
		MoBY [21]	41.4	8.80	39.6	34.1	

Table 1: Quantitative results in the main experiments. The SwinTF backbone outperforms its CNN counterpart by a non-trivial margin. Our proposed style mixing and contrastive learning ingredients consistently improve the model performance across four domains. However, adding the SSL pretraining degrades the performance considerably. Please refer to Sec. 4.4 for our discussion.

original partitions. For each domain, we take the original training and validation sets from the other three domains as the new training data. All training and validation set associated with the current domain act as its new validation set. For example, the fog domain training data consists of all publicly available RGB images and annotations of the nighttime, rain, and now domains. Namely, the model trained for the fog domain will never encounter any data in the fog domain during learning.

As for the evaluation metrics, we use the mean intersection over union (mIoU) for all experiments.

4.2 Baselines

Backbone. To verify the effectiveness of the transformer backbone, we compare the Swin Transformer vanilla version (SwinTF) [15] followed by the UperNet [20] decoder and the state-of-the-art CNN-based method HRNet [18]. Different variants of the SwinTF backbone are adopted, *e.g.*, tiny, small and base. The model capacity is: tiny $<$ small $<$ base. Please refer to [15] for more details.

SSL pretraining. We follow the implementation of SimMIM [22] for the masked signal modeling introduced in Sec. 3.1. We compare the masked signal modeling SSL with the momentum contrast (moco) [9] based SSL implemented in MoBY [21]. Both are tailored to the SwinTF backbone.

4.3 Main Results

The SwinTF-small backbone is adopted in our main experiments. We primarily construct three kinds of models: 1) the SwinTF vanilla model; 2) incorporating the style mixing and contrastive learning into the SwinTF; 3) further adding different SSL pretraining using the ACDC data. Please note that the SSL pretraining tasks use the same ACDC dataset as the final semantic segmentation tasks. Given our limited GPU resources, the pretraining tasks on the ImageNet dataset [5] are too computation-intensive to finish. Please see Sec. 4.4 for a discussion on the SSL pretraining dataset.

4.3.1 Quantitative Results

The quantitative results are shown in Table. 1. Firstly, we verify that the transformer backbone is more powerful than the state-of-the-art CNN architecture, even without further tuning. The SwinTF outperforms the HRNet consistently across all experiments. Moreover, the SwinTF with our proposed style mixing and supervised contrastive learning has considerable performance gains compared to the SwinTF vanilla method. It achieves the best performance among all the models. Notably, adding the SSL via masked signal modeling or momentum contrast does not help but degrade the performance substantially. We conjecture that this is due to the limited size of the ACDC dataset in contrast to the ImageNet dataset, the latter of which is frequently used in many SSL experiments. The SSL pre-trained models may learn some naive representations that could not generalize well for segmentation or memorize the original data. On the other hand, the non-SSL methods initiated with the ImageNet classification pre-trained networks show much better performance than our SSL variants. This indicates that the pretraining is crucial, yet the SSL pretraining on a small dataset may not be as helpful.

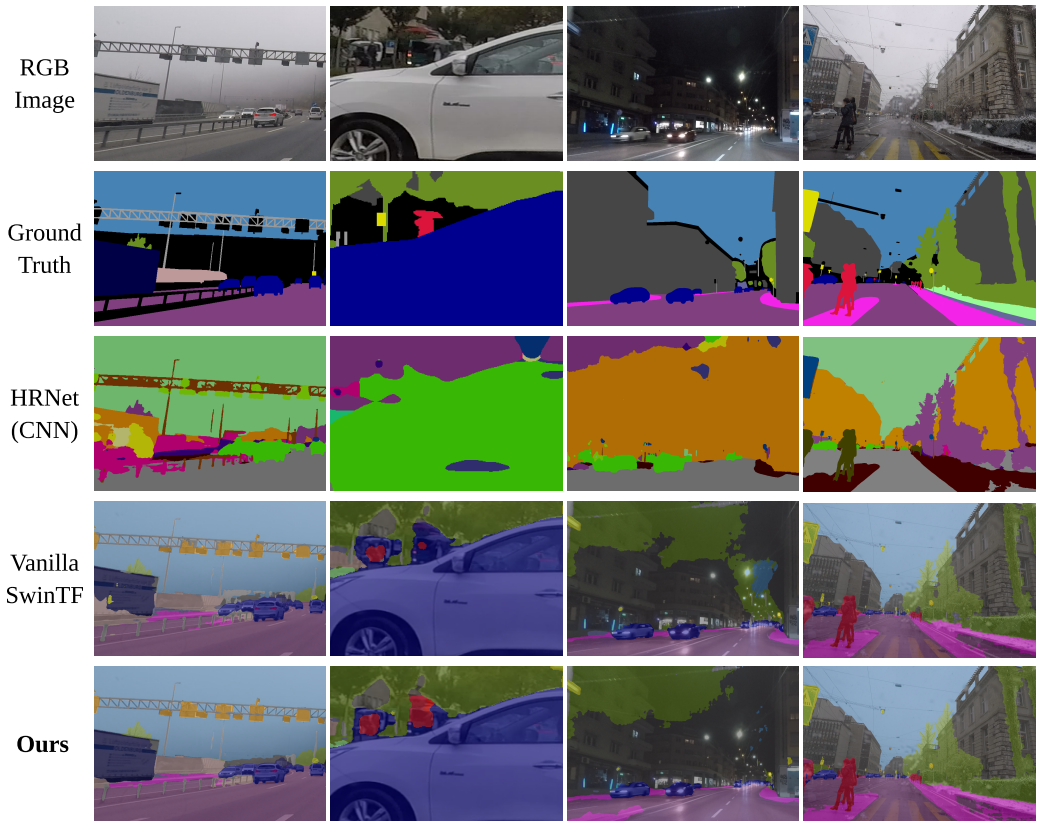

Figure 5: Qualitative results of different domain segmentation with different models

4.3.2 Qualitative Results

We get the qualitative results showing in Fig.5. We select one image from each domain and compare the results among three models. Each column represents one domain, from left to right, we have the results of fog, rain, night and snow domain respectively. From top to bottom, each row represents the original RGB image, ground truth annotation, and results from HRNet, Swin-TF and our model.

We can clearly see that HRNet hardly works. All four results from HRNet are worst. Between Swin-TF and our model, we can find that our model has a better annotation in some details. In fog domain, our model has more standard square sign boards than Swin-TF model. The edges of cars are clearer, no extra annotation around cars. In rain domain, our model can detect the small person successfully and mark it with red annotation. In night domain, both models don't work very well but our model has less wrong green annotation on the sky than the Swin-TF baseline. In the snow domain, edges of annotations are clearer and more completed.

4.4 Ablation Studies on SSL Pretraining

In this stage, we use the ImageNet dataset pretrained SSL models, obtained by the SimMIM and MoBY open code. For SimMIM method, we use the SwinTF base backbone; and for MoBY method, we use the SwinTF Tiny backbone.

Table. 2 and Table. 3 show the ablation experiment results.

Clearly, the performance degradation issue introduced by the SSL is now alleviated. This verifies our assumption about the effects of datasets on the model pretraining.

Method	Backbone	SSL	Fog	Night	Rain	Snow	Comments
SwinTF vanilla [15]	Base	None	**77.5**	**40.9**	**72.9**	**70.1**	Baseline
SwinTF**+Style Mixing+Contrast**			76.4	40.8	72.2	70.1	Ours
SwinTF**+SSL+Style Mixing+Contrast**		SimMIM [22]	74.1	37.2	70.3	65.1	**ImageNet** SSL

Table 2: Quantitative Result on SwinTF Base backbone

Method	Backbone	SSL	Fog	Night	Rain	Snow	Comments
SwinTF vanilla [15]	Tiny	None	74.2	33.6	69.1	**66.4**	Baseline
SwinTF**+Style Mixing+Contrast**			73.8	34.2	**70.2**	66.2	Ours
SwinTF**+SSL+Style Mixing+Contrast**		MoBY [21]	72.1	**36.8**	68.5	65.2	**ImageNet** SSL

Table 3: Quantitative Result on SwinTF Tiny backbone

5 Conclusion

In this paper, we propose a transformer-based segmentor that is robust to domain changes. We aim to address the domain shift issue in two phases, pre-training phase and fine-tuning phase. We pre-train the model with masked signal to learn constructive feature embeddings, and then fine-tune the model with mixed patch-wise styles and contrastive loss to learn domain-invariant pixel embeddings. Extensive experiment results demonstrate that the proposed algorithm outperforms vanilla transformer.

Our approach explores three possible ways to counteract the domain shift problem, while the proposed segmentor can still be further improved. For instance, mixing patch styles strategically based on the regions or semantic labels instead of random shuffling may have a better performance, as styles in certain regions may not make sense in other regions. In addition, We only explore style mixing with the same patch size as a transformer block, while mixing styles of larger patches may generate more meaningful feature statistics. Furthermore, contrastive learning with style mixing can also be deployed in the self-supervised learning to see if it can also benefit pre-trained model.

References

[1] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[2] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021.

[3] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *Computer Vision and Pattern Recognition*, 2021.

[4] 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.

[5] 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.

[6] 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. *Computer Vision and Pattern Recognition (cs.CV)*, 2020.

[7] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on visual transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1.

[8] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *Computation and Language (cs.CL)*, 2021.

[9] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[10] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.

[11] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019.

[12] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017.

[13] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and Ling-Yu Duan. Uncertainty modeling for out-of-distribution generalization. *International Conference on Learning Representations:2202.03958*, 2022.

[14] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. *Computer Vision and Pattern Recognition (cs.CV)*, 2021.

[15] 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.

[16] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *Machine Learning (cs.LG):1807.03748*, 2018.

[17] 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.

[18] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020.

[19] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring Cross-Image Pixel Contrast for Semantic Segmentation. January 2021.

[20] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 418–434, 2018.

[21] Zhenda Xie, Yutong Lin, Zhuliang Yao, Zheng Zhang, Qi Dai, Yue Cao, and Han Hu. Self-supervised learning with swin transformers. *Computer Vision and Pattern Recognition (cs.CV):2105.04553*, 2021.

[22] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. *Computer Vision and Pattern Recognition (cs.CV):2111.09886*, 2021.

[23] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *International Conference on Learning Representations (ICLR)*, 2021.