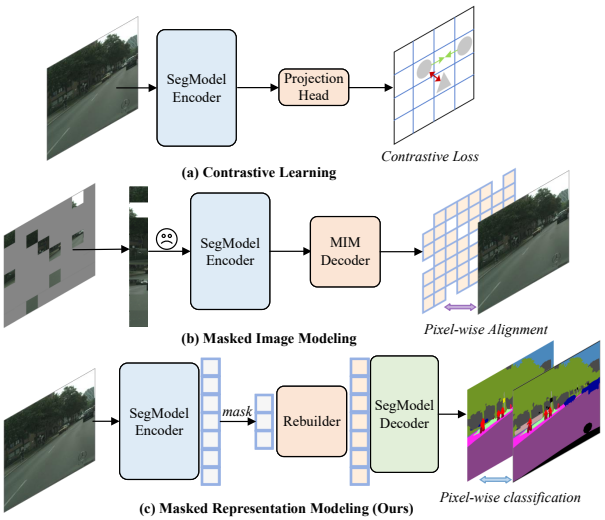
Masked Representation Modeling for Domain-Adaptive Segmentation

Wenlve Zhou[1] Zhiheng Zhou[1*] Tiantao Xian[1] Yikui Zhai[2] Weibin Wu[3] Biyun MA[1]

[1]South China University of Technology
[2]School of Electronic and Information Engineering, Wuyi University
[3]South China Agricultural University

Abstract

Unsupervised domain adaptation (UDA) for semantic segmentation seeks to transfer models from a labeled source domain to an unlabeled target domain. While auxiliary self-supervised tasks such as contrastive learning have enhanced feature discriminability, masked modeling remains underexplored due to architectural constraints and misaligned objectives. We propose Masked Representation Modeling (MRM), an auxiliary task that performs representation masking and reconstruction directly in the latent space. Unlike prior masked modeling methods that reconstruct low-level signals (e.g., pixels or visual tokens), MRM targets high-level semantic features, aligning its objective with segmentation and integrating seamlessly into standard architectures like DeepLab and DAFormer. To support efficient reconstruction, we design a lightweight auxiliary module, Rebuilder, which is jointly trained with the segmentation network but removed during inference, introducing zero test-time overhead. Extensive experiments demonstrate that MRM consistently improves segmentation performance across diverse architectures and UDA benchmarks. When integrated with four representative baselines, MRM achieves an average gain of +2.3 mIoU on GTA → Cityscapes and +2.8 mIoU on Cityscapes → Synthia, establishing it as a simple, effective, and generalizable strategy for unsupervised domain-adaptive semantic segmentation.



Properties	Input aligns with main task	Encoder training	Decoder training
CL	✓	✓	×
MIM	×	×	×
MRM (ours)	✓	✓	✓

Figure 1. Comparison of three auxiliary tasks for UDA segmentation. (a) Contrastive Learning (CL) uses contrastive loss for feature alignment but does not train the decoder, limiting end-to-end optimization. (b) Masked Image Modeling (MIM) reconstructs masked components but disrupts the segmentation pipeline, reducing compatibility with certain architectures. (c) Masked Representation Modeling (MRM) performs masking and reconstruction in latent space, aligns with the segmentation task, remains compatible with diverse architectures, and improves performance without inference overhead, as the Rebuilder is used only during training.

1. Introduction

Deep learning has achieved significant success in computer vision, including tasks such as semantic segmentation [32, 33, 41]. However, these models often struggle when faced with domain shift—a mismatch between training and testing data distributions—which can lead to notable performance degradation [49]. Addressing this issue by collecting and annotating new target domain data is resource-intensive and time-consuming, particularly for dense pixel-level annotations in semantic segmentation [11, 39]. Unsupervised domain adaptation (UDA) [57] offers a practical alternative, enabling models to utilize labeled source domain data and unlabeled target domain data to improve performance without additional annotation efforts.

*Corresponding author: zhouzh@scut.edu.cn
Code available at https://github.com/Wenlve-Zhou/MRM

Auxiliary tasks provide a promising direction for enhancing UDA-based segmentation models [8, 21, 44, 51]. These tasks, typically designed to require no manual annotations and to integrate seamlessly with existing architectures, enrich representations without disrupting the primary network. Among them, contrastive learning [9, 15, 34, 40] has shown strong potential by improving feature discrimination: pulling similar samples closer while pushing dissimilar ones apart (Figure 1(a)). This makes it particularly useful for UDA segmentation, where distinguishing challenging categories is critical [6, 8, 26, 27, 51]. However, another self-supervised technique—masked image modeling (MIM)—has seen limited exploration in the context of UDA segmentation. MIM methods, such as masked autoencoders (MAE) [16], train models to reconstruct occluded regions, fostering a better understanding of global context and scene structure. While this approach seems suitable for UDA segmentation by encouraging models to capture broader scene context, its adoption is limited. We speculate that this mainly stems from two main challenges:

(i) *Input structure constraints:* MIM modifies the input structure by masking image patches (Figure 1(b)), complicating its application to segmentation networks like DeepLab [24] or DAFormer [19].

(ii) *Optimization conflicts:* MIM methods focus on element-wise reconstructing occluded patches, which may introduce conflicts with the optimization objectives of domain adaptive segmentation tasks.

To address these challenges, we propose Masked Representation Modeling (MRM), a simple yet effective auxiliary task for unsupervised domain-adaptive semantic segmentation (Figure 1(c)). Unlike image-level masking methods, MRM performs representation masking and reconstruction in latent space, avoiding disruptions to the input and making it compatible with a wide range of architectures, including CNNs [24] and Transformers [19]. More importantly, MRM aligns its optimization objective with the primary segmentation task by using the segmentation decoder to perform pixel-wise classification on reconstructed representation, rather than enforcing alignment in pixel space [16, 50], thereby reducing conflicts between auxiliary and main tasks and uniquely enhancing the decoder—a benefit not typically offered by contrastive learning [9, 15]. By encouraging the network to predict missing latent representations, MRM implicitly improves feature robustness and cross-domain generalization, which is particularly critical in scenarios with large domain shifts.

To facilitate representation reconstruction, we introduce a lightweight, asymmetric Rebuilder module inspired by masked image modeling [2, 16, 48, 50]. The Rebuilder is jointly trained with the segmentation network and removed after training, introducing zero additional inference overhead. Extensive experiments demonstrate that MRM consistently improves segmentation performance across diverse architectures and benchmarks. For instance, when combined with four representative baselines, it achieves average gains of +2.3 mIoU on GTA→Cityscapes and +2.8 mIoU on Cityscapes→Synthia, highlighting MRM as a generalizable, plug-and-play strategy for enhancing UDA segmentation. These results suggest that representation-level masked modeling can serve as a versatile auxiliary objective, complementing existing adaptation techniques without requiring architectural modifications or complex training procedures.

2. Related Work

Semantic segmentation, a cornerstone of computer vision, has seen significant progress with deep learning. Fully Convolutional Networks (FCNs) [31] enabled pixel-level understanding, but challenges like small object delineation and complex scenes persist. The encoder-decoder architecture improved segmentation with innovations like skip-connections [37] for feature fusion and dilated convolutions [7] to expand receptive fields. While CNN-based methods dominated early research, Transformers [10, 23, 52, 56, 58] have advanced global context modeling. Despite their performance, Transformers' self-attention mechanism is computationally intensive for high-resolution images [45]. Xie *et al.* [52] mitigated this by downsampling key and value components in self-attention.

Unsupervised domain adaptive segmentation addresses the challenge of adapting segmentation models to new domains without target annotations. Common approaches include adversarial training [18, 44], self-training [43, 47, 59], and efficient architecture design [19, 22]. Beyond these, auxiliary tasks have been introduced to enhance feature representation: contrastive learning improves backbone features [1, 8, 51], while depth estimation aids scene understanding [21, 53]. Though effective, these methods typically focus on the encoder and may conflict with the main task. Our method, Masked Representation Modeling (MRM), overcomes these limitations by training the entire model, aligning encoder and decoder under the same objective to reduce potential conflicts.

Masked image modeling (MIM) learns visual representations by reconstructing masked image regions [16, 17, 30, 50, 54]. Originating from image inpainting, Context Encoders [35] used convolutional networks to predict missing regions. Building on this idea, ViT-based approaches [4, 13, 16] extended masking to self-supervised learning, from patch prediction to discrete token modeling, while ConvNeXtV2 [50] demonstrated its effectiveness for convolutional architectures. Despite the success of MIM in representation learning, masking input patches disrupts the data-processing pipeline, making methods like MAE [16] difficult to integrate into segmentation frameworks such as

DeepLab [24] and DAFormer [19]. To address this, we propose MRM, which performs masking and reconstruction in the latent space, preserving the input paradigm and supporting diverse segmentation architectures.

3. Method

3.1. Preliminary

Unsupervised domain adaptive segmentation aims to train a neural network using labeled source domain data $D_s = \{(x_k^s, y_k^s)\}_{k=1}^{n_s}$ to perform well on a target domain $D_t = \{(x_k^t)\}_{k=1}^{n_t}$, without access to target labels. In semantic segmentation model (SegModel), typically consists of an Encoder $E(\cdot)$ and a Decoder $D(\cdot)$. Simply training the network using pixel-wise cross-entropy on the source domain can be formulated as:

$$\mathcal{L}_{sup} = -\sum_{i=1}^{H}\sum_{j=1}^{W}\sum_{c=1}^{C} y_{ijc}^s \log D(E(x^s))_{ijc} \tag{1}$$

where H and W represent the image's height and width respectively, while C signifies the number of categories in the UDA task.

However, a model trained solely on the source domain often experiences a performance decline when applied to a different domain. To address this issue, UDA methods leverage unlabeled target domain images to adapt the network. To achieve this, an additional unsupervised loss $\mathcal{L}_{uda}$ is introduced into the optimization process.

$$\mathcal{L}_{overall} = \mathcal{L}_{sup} + \mathcal{L}_{uda} \tag{2}$$

Masked image modeling is a self-supervised learning paradigm designed to train neural networks by reconstructing masked portions of an input image. The input image x is reshaped into non-overlapping patches $p = \{p_i\}_{i=1}^{N_l}$. MIM constructs a random mask $M^{mim} \in \{0,1\}^{N_l}$ to indicate the masked patches, where $M_i^{mim} = 1$ corresponds to the patches that are masked. Only the visible patches $p^v = \left\{p_i \mid M_i^{mim} = 0\right\}_{i=1}^{N_v}$ are fed into the encoder and are mapped to potential features, after which a lightweight decoder D_{mim} is introduced to reconstruct the pixel values of the invisible patches.

$$\mathcal{L}_{mim} = \sum_{i=1}^{N_l} M_i^{mim} \cdot \left(D_{mim}(E(p_i^v)) - p_i\right)^2 \tag{3}$$

Incorporating MIM as an auxiliary task poses challenges since typical UDA segmentation architectures [19, 24] are designed to process complete images rather than visible patches.

3.2. Masked Representation Modeling

Masked Representation Modeling (MRM) is a simple approach designed to address the limitations of mainstream MIM methods when used as an auxiliary task. Unlike MIM, MRM eliminates the need for the SegModel encoder $E(\cdot)$ to process partially visible signals. Instead, it reconstructs the latent representation from the partially visible features provided to the encoder. The SegModel decoder $D(\cdot)$ then processes the reconstructed representation, and the model is trained through a pixel-wise classification task.

$$\mathcal{L}_{mrm} = -\sum_{i=1}^{H}\sum_{j=1}^{W}\sum_{c=1}^{C} \tilde{y}_{ijc} \log D(R(E(x^t)))_{ijc} \tag{4}$$

Here, $\tilde{y}$ is the pseudo label [25] of target image, and $R(\cdot)$ represents the Rebuilder (*refer to next section for detailed information*), whose role is analogous to the lightweight decoder in MIM [16, 50]. Both are responsible for reconstructing the original signal from partially visible signals. When the MRM task is used as an auxiliary task for unsupervised domain adaptive segmentation, the overall optimization objective can be expressed as:

$$\mathcal{L}_{overall} = \mathcal{L}_{sup} + \mathcal{L}_{uda} + \lambda \mathcal{L}_{mRm} \tag{5}$$

where λ is the trade-off in masked representation modeling. MRM can be easily integrated with existing UDA methods.

3.3. Rebuilder

The Rebuilder $R(\cdot)$ primarily serves to randomly mask out representation regions from SegModel encoder and then reconstruct them. Its design is largely consistent with the decoder [2, 5, 16, 48] used in MIM. After the semantic segmentation model has been fully trained, the Rebuilder is removed, ensuring it does not impact the inference process of the original model. The pipeline of the Rebuilder is shown in Figure 2, and the following sections will discuss the specific design details. See **Supplementary Material** for further analysis.

Representation embedding. Since different encoders [19, 24] produce representation at varying scales, it is essential to ensure that subsequent transformer blocks can efficiently process the features from different encoders in a consistent manner. Thus, the input feature $f^t = E(x^t)$, $f^t \in \mathbb{R}^{C \times H \times W}$ is scaled to $C' \times H' \times W'$ to obtain embedding representation $\tilde{f}^t$. For the channel dimension, a linear mapping layer is applied for rescaling. After processing the channel dimension, bilinear interpolation is used to resize the input representation to the target spatial dimensions when the input and target spatial dimensions are not aligned.

Masking. Based on the shape of the features obtained after representation embedding, a binary mask $M \in \mathbb{R}^{H' \times W'}$ is generated through uniform random sampling [16]. We apply the generated mask to perform masked out operation

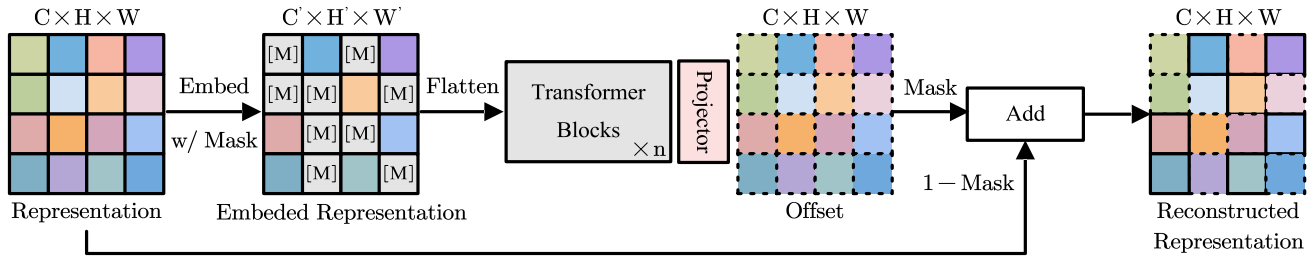
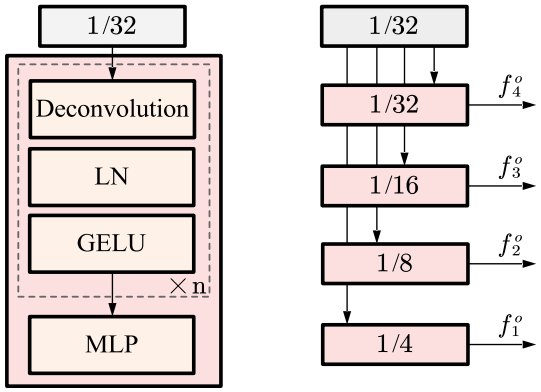


Figure 2. The pipeline of Rebuilder. The Rebuilder is designed to randomly mask out representation from the encoder and reconstruct the masked component. It first scales the encoder representation along both spatial and channel dimensions, and then applies random masking to remove a subset of these representation. Subsequently, the masked representation are passed through several Transformer blocks and a projector to generate reconstructed representation, which are input to the decoder for model training. [M] is a learnable token.



(a) Projector Details **(b) Multi-Scale Projector**

Figure 3. An overview of the projector. The representation from the Transformer are reshaped and fed into the projector, which uses several transposed convolutions to generate features at different scales. (a) The projector details and (b) the multi-scale projector.

on the embedding representation, and fill the corresponding positions of the removed representation with a mask token. Each mask token [12] is a shared, learnable vector that represents the presence of a missing patch to be predicted. The processed representation are reshaped to $(H'W') \times C'$ and fed into the subsequent parts of the architecture.

Architecture. Since the mask token lacks positional information, following works [2, 5, 16, 48] the reshaped representation are first added with absolute positional embeddings [45]. A series of Transformer blocks [13] are then applied to process these representation. Experiment demonstrate that MRM achieves performance gains with only a minimal number of blocks (e.g., $n = 1$ or $n = 2$), thus avoiding significant computational overhead, details in Table 2(b). In addition, a projector is used to map the Transformer's output to the original representation dimensions. The projector consists of transposed convolution layers that rescale both spatial and channel dimensions to align with the original representation, as shown in Figure 3(a).

After obtaining the offset f^o from the Projector's output, the mask is resized to the original feature map size to obtain $M^s \in \mathbb{R}^{H \times W}$. Based on M^s, the offset are fused with the original representation f^t to obtain the final reconstructed representation f^r, where broadcasting is applied due to the channel dimension mismatch between the mask and feature tensors.

$$f^r = M^s \odot f^o + (1 - M^s) \odot f^t \tag{6}$$

where $\odot$ is Hadamard product.

Implemented with multi-scale model. For multi-scale models (e.g., DAFormer [19]), we do not instantiate a separate Rebuilder at each stage, as such a naive design would introduce substantial computational and memory overhead due to the Transformer components. Instead, we leverage only the representation from the final encoder stage for embedding and Transformer processing. As illustrated in Figure 3(b), we introduce an individual upsampling operation for each target scale to project the final-stage representation into multi-scale features. In this way, the reconstructed features at different resolutions are generated directly from the same high-level representation, rather than by applying Transformer-based rebuilding at every stage. This design preserves the multi-scale property of hierarchical architectures while avoiding redundant computation. The feasibility of this design is supported by ViTDet [28], which shows that multi-scale features can be derived from final-stage representations through simple upsampling operations. Consequently, for architectures such as DAFormer [19], our MRM introduces no additional Transformer overhead while maintaining effective reconstruction across scales. Formally, given a hierarchical representation f_i^t, where $i \in 1, 2, 3, 4$, the reconstructed representation is computed as

$$f_i^r = M_i^s \odot f_i^o + (1 - M_i^s) \odot f_i^t. \tag{7}$$

4. Experiments

In this section, we focus on experimenting with two popular benchmarks: GTA → Cityscape and Synthia → Cityscape. First, we introduce the datasets along with the implementation details. Then, we compare our approach with state-of-the-art (SoTA) methods. Next, we explore the performance impact under various parameter settings through ablation studies. Finally, we present qualitative analysis to further illustrate the effectiveness of our method.

4.1. Experiment Setups

Dataset. The GTA [36] dataset comprises 24,966 synthetic images featuring pixel-level semantic annotations. These images are generated within the open-world environment of "Grand Theft Auto V", all captured from the perspective of a vehicle navigating the streets of American-style virtual cities. This dataset encompasses 19 semantic classes that align with those found in the Cityscapes dataset.

SYNTHIA [38] constitutes a synthetic urban scene dataset. We opt for its subset known as "SYNTHIA-RAND-CITYSCAPES", which shares 16 common semantic annotations with Cityscapes. Specifically, we utilize a total of 9,400 images, each with a resolution of 1280×760, sourced from the SYNTHIA dataset.

Cityscapes [11] is a dataset featuring real urban scenes captured across 50 cities in Germany and neighboring regions. The dataset includes meticulously annotated images, comprising 2,975 training images, 500 validation images, and 1,525 test images, all at a resolution of 2048×1024 pixels. Each pixel within these images is classified into one of 19 distinct categories.

Baseline methods. We select popular UDA segmentation methods such as DACS [43], DAFormer [19], HRDA [22], and MIC [20] as baselines to validate the performance improvement of MRM on these baselines. We also compare our method with popular approaches such as QuadMix [55], PiPa [8], and GANDA [29].

Implementation details. To ensure fairness and reproducibility, the hyperparameters related to the Baselines are kept consistent with those in the original paper. For network architecture, we utilize the DeepLab-V2 [24] with ResNet101 [14] and DAFormer [19] with MiT-B5 [52] as the architecture.

For Masked Representation Modeling, we chose a trade-off λ value of 1.0. Regarding the design of the Rebuilder, the number of transformer blocks is set to 2, and the embedding dimension is set to 512. In the representation embedding section, we choose to scale the spatial and channel dimensions of the feature map, specifically setting $H' = W' = 16$ and $C' = 512$. We set the masking ratio to 40%. The learning rate of 2×10^{-4} is employed with the Rebuilder. For the HRDA head [22], the MRM is trained using both the fused multi-scale predictions and the detail crop predictions, following the methodology described in the original paper. The results are averaged over 3 random seeds. All the experiments are conducted with 1x NVIDIA GTX 3090 with 24G RAM and the PyTorch framework is implemented to perform our experiments.

4.2. Comparisons with State-of-the-Arts UDA Methods

We evaluate our method against SoTA UDA approaches on two widely used benchmarks: GTA5 → Cityscapes and SYNTHIA → Cityscapes, using a variety of base learners. As shown in Table 1, integrating the proposed MRM module leads to consistent performance improvements.

On the GTA5 → Cityscapes benchmark, MRM yields notable gains: +3.8 mIoU for DACS [43], +2.0 mIoU for DAFormer [19], +1.6 mIoU for HRDA [22], and +1.6 mIoU for MIC [20]. The improvement is particularly pronounced in fine-grained categories such as traffic sign, rider, and motorbike, suggesting that MRM may enhance the decoder's capacity for high-level semantic discrimination. In particular, the MIC [20] + MRM combination achieves an mIoU of 77.5, which surpasses previous state-of-the-art results by a margin of +1.4.

On the more challenging SYNTHIA → Cityscapes benchmark, MRM consistently improves performance: +7.5 mIoU for DACS [43], +1.7 for DAFormer [19], +1.3 for HRDA [22], and +0.8 for MIC [20]. These results suggest that MRM generalizes well across different adaptation strategies and datasets.

Overall, the findings indicate that MRM serves as a model-agnostic, plug-and-play auxiliary task that can complement existing UDA pipelines by more effectively aligning feature learning with semantic segmentation objectives.

4.3. Ablation Study

In the ablation study, we adopt DACS [43] as the baseline and conduct experiments on the GTA → Cityscapes benchmark using both DeepLab-V2 [24] with ResNet101 [14] and DAFormer [19] with MiT-B5 [52] architectures to ensure consistent and generalizable analysis across different designs.

Masking ratio. Figure 4 shows that MRM performs best at a masking ratio of 40%, reaching 55.9 mIoU (+3.8 over baseline). Unlike MAE [16] and ConvNeXtV2 [50], which favor high masking ratios (60–75%), MRM benefits from lower ratios. Excessive masking reduces the diversity of reconstructed representation and harms semantic consistency, especially since MRM's visible tokens are processed by fewer Transformer blocks than in full MIM setups.

Rebuilder design. The Rebuilder is designed to be lightweight, ensuring MRM's auxiliary training introduces minimal overhead. Results in Tables 2(a–b, d) show that increasing the embedding dimension and Transformer depth

Method	Road	S.walk	Build.	Wall	Fence	Pole	Tr.Light	Sign	Veget.	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	M.bike	Bike	mIOU
GTA → Cityscape																				
PiPa [8]	96.1	72.0	90.3	56.6	52.0	55.1	61.8	63.7	90.8	52.6	93.6	74.3	43.6	93.5	78.4	84.2	77.3	59.9	66.7	71.7
GANDA [29]	96.5	74.8	91.4	61.7	57.3	59.2	65.4	68.8	91.5	49.9	**94.7**	79.6	54.8	94.1	81.3	86.8	74.6	64.8	68.2	74.5
QuadMix [55]	97.5	**80.9**	91.6	62.3	**57.6**	58.2	64.5	71.2	91.7	52.3	94.3	80.0	55.9	**94.6**	86.3	90.5	82.3	65.1	68.1	76.1
DACS [43]	89.9	39.7	87.9	30.7	39.5	38.5	46.4	52.8	88.0	44.0	88.8	67.2	35.8	84.5	45.7	50.2	0.0	27.3	34.0	52.1
w/ MRM	94.7	68.3	87.6	38.1	27.4	42.0	51.9	59.2	86.9	45.0	87.7	66.3	32.3	89.6	53.9	56.6	0.0	33.2	42.1	55.9
DAFormer [19]	95.7	70.2	89.4	53.5	48.1	49.6	55.8	59.4	89.9	47.9	92.5	72.2	44.7	92.3	74.5	78.2	65.1	55.9	61.8	68.3
w/ MRM	96.8	76.3	89.5	55.7	50.3	50.9	58.2	62.2	90.0	50.1	91.1	73.8	48.2	92.3	77.3	80.4	71.6	56.3	64.0	70.3
HRDA [22]	96.4	74.4	91.0	61.6	51.5	57.1	63.9	69.3	91.3	48.4	94.2	79.0	52.9	93.9	84.1	85.7	75.9	63.9	67.5	73.8
w/ MRM	97.0	78.3	90.9	59.5	52.8	60.4	66.6	72.4	**91.9**	51.8	94.3	79.3	55.5	94.4	84.4	87.9	82.7	65.4	68.2	75.4
MIC [20]	97.4	80.1	91.7	61.2	56.9	59.7	66.0	71.3	91.7	51.4	94.3	79.8	56.1	**94.6**	85.4	90.3	80.4	64.5	68.5	75.9
w/ MRM	**98.3**	80.4	**92.6**	**62.7**	57.0	**62.3**	**69.1**	**74.3**	91.8	**53.5**	**94.7**	**81.1**	**56.6**	94.1	**87.2**	**91.6**	**85.4**	**66.3**	**71.3**	**77.5**
Synthia → Cityscape																				
PiPa [8]	87.9	48.9	88.7	45.1	4.5	53.1	59.1	58.8	87.8	–	92.2	75.7	49.6	88.8	–	53.5	–	58.0	62.8	63.4
GANDA [29]	89.1	50.6	**89.7**	**51.4**	6.7	59.4	**66.8**	57.7	86.7	–	93.8	80.6	56.9	**90.7**	–	64.8	–	62.6	65.0	67.0
QuadMix [55]	88.1	51.2	88.9	46.7	7.9	58.6	64.7	**63.7**	88.1	–	93.9	81.3	56.6	90.3	–	**66.9**	–	66.8	**66.0**	67.5
DACS [43]	80.6	25.1	81.9	21.5	2.9	37.2	22.7	24.0	83.7	–	90.8	67.6	38.3	82.9	–	38.9	–	28.5	47.6	48.3
w/ MRM	83.5	44.2	84.9	21.5	3.4	41.6	47.2	52.3	83.7	–	86.5	68.6	42.1	84.1	–	51.2	–	40.4	58.2	55.8
DAFormer [19]	84.5	40.7	88.4	41.5	6.5	50.0	55.0	54.6	86.0	–	89.8	73.2	48.2	87.2	–	53.2	–	53.9	61.7	60.9
w/ MRM	88.4	51.4	88.8	41.2	**8.4**	50.2	55.9	52.9	85.8	–	88.5	73.0	47.6	87.2	–	63.1	–	57.8	61.4	62.6
HRDA [22]	85.2	47.7	88.8	49.5	4.8	57.2	65.7	60.9	85.3	–	92.9	79.4	52.8	89.0	–	64.7	–	63.9	64.9	65.8
w/ MRM	**90.6**	**55.5**	88.3	49.8	7.0	57.8	65.6	56.6	87.9	–	93.9	79.6	53.2	89.1	–	65.1	–	**67.8**	**66.0**	67.1
MIC [20]	86.6	50.5	89.3	47.9	7.8	59.4	66.7	63.4	87.1	–	94.6	81.0	58.9	90.1	–	61.9	–	67.1	64.3	67.3
w/ MRM	87.5	53.2	**89.7**	48.7	7.8	**61.0**	66.7	63.1	**88.6**	–	**94.7**	**81.8**	**59.7**	90.1	–	64.3	–	**67.8**	65.3	**68.1**

Table 1. Comparison with state-of-the-art methods for UDA. The best result in each metric column is marked bold. The results for MRM are averaged over 3 random seeds

Dim	128	256	512	768
mIOU	54.7	55.1	**55.9**	53.6

(a) Embedding dimensions of Transformer blocks in the Rebuilder.

Blocks	1	2	4	8
mIOU	55.4	**55.9**	54.4	52.9

(b) Number of Transformer blocks used in the Rebuilder.

Case	mIOU
Baseline	52.1
Masking	51.9
Masking + Rebuilding	**55.9**

(c) Effect of masking and rebuilding on performance.

$H' = W'$	8	16	32	64
mIOU	55.6	**55.9**	54.1	OOM

(d) Spatial dimensions of the rescaled representation. OOM marks out of memory.

Case	mIOU
Baseline	52.1
pixel rec. (w/ norm) [16]	51.8
feature rec. (w/ teacher) [5]	53.5
feature rec. (w/o teacher) [5]	53.7
pixel cls.	**55.9**

(e) Training objective. Consistent objectives boost performance.

λ	0.1	0.5	1.0	2.0	10.0
mIOU	54.7	55.8	**55.9**	55.4	52.1

(f) Influence of the loss weight λ on performance.

Table 2. Ablation study of MRM using DeepLab-V2 [24] with ResNet-101 [14] on the GTA → Cityscape benchmark with DACS [43].

improves performance up to a point (best with dim=512, 2 blocks, $H' = W' = 16$). Beyond this scale, performance drops due to instability when training deep Transformers atop pre-trained backbones. This suggests that moderate capacity stabilizes optimization while preserving auxiliary benefits, and future work could explore stronger yet stable Rebuilder designs.

Masking vs. rebuilding. To disentangle contributions,

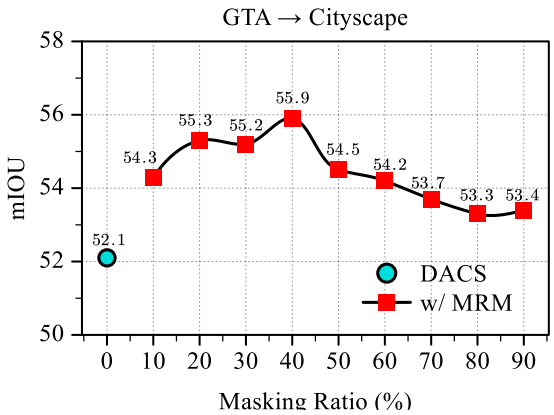


Figure 4. Masking ratio. The optimal performance enhancement is achieved when the masking ratio is adjusted to 40%.

we separately test masking "$(1 - M^s) \odot f^t$" and rebuilding "$M^s \odot f^o$". As shown in Table 2(c), masking alone slightly harms performance (–0.2 mIoU), indicating that feature-space masking causes irreversible semantic loss. Adding the rebuilding branch restores lost semantics and yields significant gains, confirming that reconstruction is essential to effective representation regularization.

Training objective. Table 2(e) compares several reconstruction objectives. Pixel-level regression (as in MAE [16]) underperforms (–0.3 mIoU) due to its low-level focus. Feature reconstruction using teacher [5] or student features [5] offers minor gains. In contrast, directly applying pixel-wise classification with cross-entropy loss yields the best result, emphasizing that auxiliary supervision should be task-aligned with segmentation.

Trade-off. Table 2(f) shows that MRM's performance is stable across a wide range of weighting factors. The best results are obtained with a trade-off of 1.0, while overly large weights (e.g., 10) slightly degrade performance, indicating that MRM requires little hyperparameter tuning.

MRM on different domain. In our main setup, MRM is applied to target-domain images for auxiliary training. Here, we examine its effect when applied to the source domain using DACS [43] and DAFormer [19]. As shown in Table 3, performance gains arise only from target-domain MRM, while applying it to both domains brings no benefit and may even reduce mIOU. This result indicates that MRM primarily aids adaptation by refining target-domain features. In contrast, source-side reconstruction biases representations toward the source distribution, weakening domain alignment. Thus, effective masked modeling for UDA should emphasize target-domain reconstruction as an adaptive regularizer rather than a generic self-supervised task.

Ablation on encoder and decoder training in MRM. Unlike contrastive-based auxiliary tasks [8, 51] that enhance only the encoder, MRM jointly optimizes both encoder and decoder, strengthening the entire segmentation pipeline. We conduct ablations using DACS [43] and DAFormer [19], freezing either component during MRM training. As shown in Table 4, performance drops notably when either part is fixed, confirming that MRM's effectiveness relies on joint encoder–decoder optimization. This highlights a promising direction for designing auxiliary tasks that supervise the full network rather than isolated components.

MRM Domain	DACS [43]	DAFormer [19]
–	52.1	68.3
Source	52.9 (+0.8)	68.2 (-0.1)
Target	55.9 (+3.8)	70.3 (+2.0)
Source + Target	55.2 (+3.1)	69.0 (+0.7)

Table 3. MRM Performance across Source and Target Domains on the GTA→Cityscapes Benchmark.

Encoder	Decoder	DACS [43]	DAFormer [19]
		52.1	68.3
✔		54.9 (+2.8)	69.3 (+1.0)
	✔	54.2 (+2.1)	69.1 (+0.8)
✔	✔	55.9 (+3.8)	70.3 (+2.1)

Table 4. Component-wise Performance Analysis of MRM Training on the GTA→Cityscapes Benchmark.

Further architectures. To assess generalization, we extend MRM to diverse encoder–decoder combinations and UDA methods, including ResNet-50/101 [14], MiT-B2/B3 [52], and decoders such as DeepLab-V3+ [7]. We also evaluate with DACS [43] and MIC [20] to cover different adaptation paradigms. As summarized in Table 5, MRM consistently improves mIoU across all configurations, with gains persisting as model capacity increases—demonstrating its plug-and-play nature and scalability across architectures.

Encoder	Decoder	UDA	w/o MRM	w/ MRM
RN50 [14]	DLV2 [24]	DACS [43]	52.0	55.1 (+3.1)
RN101 [14]	DLV3+ [7]	DACS [43]	54.7	59.3 (+4.6)
RN101 [14]	DLV2+ [24]	MIC [20]	64.2	67.1 (+2.9)
MiT-b2 [52]	DAFH [19]	DAFormer [19]	64.2	66.3 (+2.1)
MiT-b3 [52]	DAFH [19]	MIC [20]	73.6	75.8 (+2.2)

Table 5. Various UDA Methods with Guidance Training for mIOU Improvement on the GTA→Cityscapes Benchmark.

4.4. Qualitative Analysis

To ensure fair comparison, we evaluate all methods under a unified DeepLabV2–ResNet101 framework and visualize

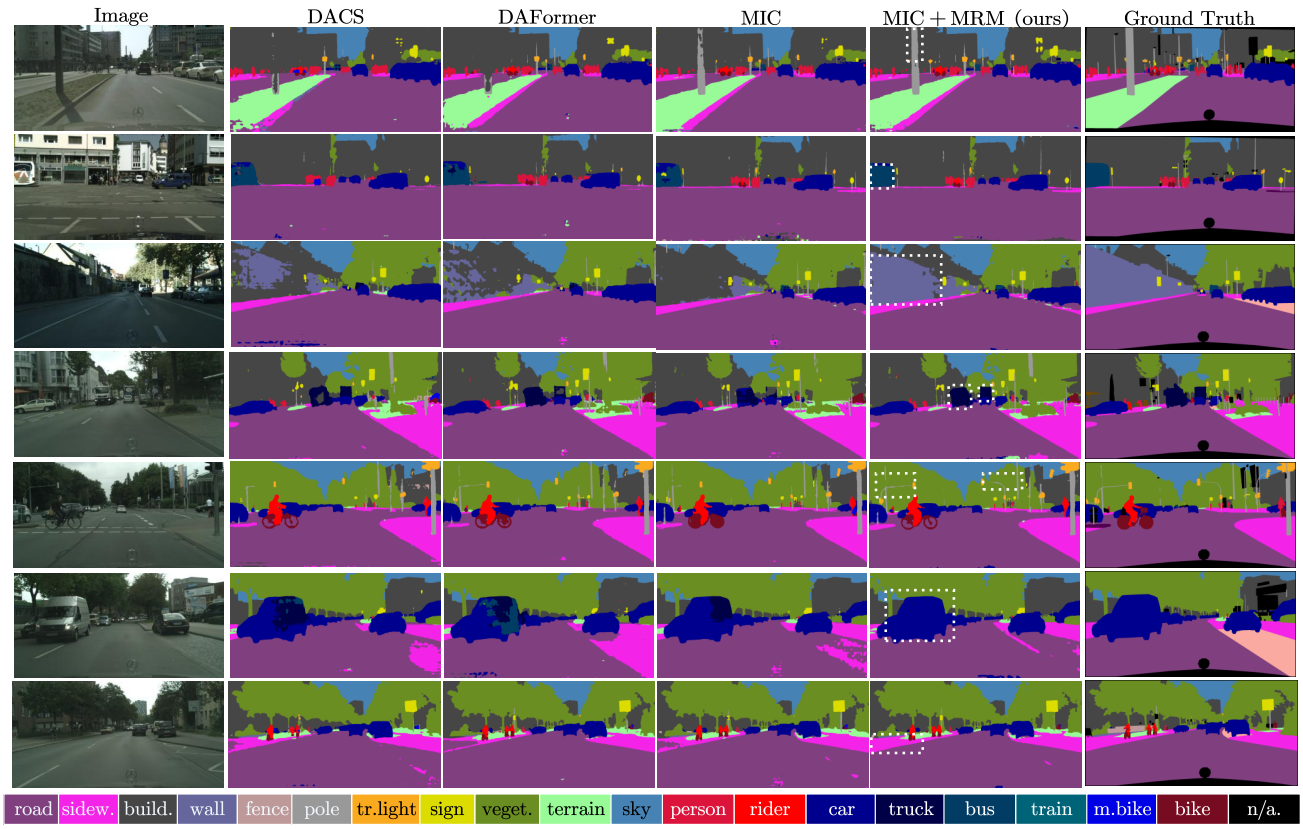


Figure 5. Qualitative comparison of MRM with previous methods on GTA → Cityscapes. To ensure a fair comparison and to demonstrate MRM's capability in contextual semantic consistency and long-range dependency modeling, we uniformly adopt the DeepLabv2 [24] with ResNet-101 [14] architecture.

results of MRM integrated with MIC. As shown in Figure 5, MRM produces cleaner and more coherent predictions than prior UDA approaches, revealing two key observations.

Contextual semantic consistency. Existing UDA models often produce fragmented or contextually inconsistent regions—such as broken "wall" structures or mixed vehicle categories—due to missing target-domain supervision. MRM alleviates these issues by enforcing feature-level reconstruction, which strengthens local continuity and preserves structural semantics across class boundaries. This indicates that auxiliary reconstruction implicitly regularizes spatial context, promoting stable domain alignment and preventing label noise from propagating through pseudo-supervised training.

Long-range dependency modeling. While MIC enhances local context, its CNN backbone restricts global reasoning. The Transformer-based Rebuilder enables MRM to capture long-range dependencies even within convolutional architectures, allowing better separation of visually similar but semantically distinct regions (e.g., "wall" vs. "building"). Beyond accuracy gains, this suggests that MRM transfers a global awareness prior to the base network—encouraging more topology-consistent segmentation and improving robustness under severe appearance shifts.

5. Conclusion

We propose Masked Representation Modeling (MRM), a novel auxiliary task for unsupervised domain adaptive semantic segmentation. Unlike conventional masked image modeling, MRM operates directly in the latent space, avoiding architectural conflicts and task misalignment. By coupling representation reconstruction with the pixel-wise segmentation objective and involving the decoder in auxiliary supervision, MRM strengthens the entire segmentation pipeline without additional inference cost. Extensive experiments across multiple benchmarks and architectures validate its generalizability and effectiveness. Our analyses further reveal that representation reconstruction not only aids domain alignment but also enhances decoder regularization, improving the consistency of cross-domain representations. We hope this study provides insights into integrating task-aligned masked modeling as an effective auxiliary signal for domain adaptation.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (32572189), the Guangdong Basic and Applied Basic Research Foundation (2024A1515140111, 2026A1515011831, 2026A1515010362), the Guangzhou Science and Technology Project (2023B01J0011), the Shaoguan Science and Technology Project (230316116276286), the Foshan Science and Technology Project (2220001018608), the Zhuhai Science and Technology Project (2320004002668), and the Zhongshan Science and Technology Project (2024A1010).

References

[1] Maregu Assefa, Muzammal Naseer, Iyyakutti Iyappan Ganapathi, Syed Sadaf Ali, Mohamed L Seghier, and Naoufel Werghi. Dycon: Dynamic uncertainty-aware consistency and contrastive learning for semi-supervised medical image segmentation. In *CVPR*, 2025. 2

[2] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, 2023. 2, 3, 4

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. In *ICLR*, 2022. 2

[5] Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024. 3, 4, 6, 7

[6] Jingkun Chen, Changrui Chen, Wenjian Huang, Jianguo Zhang, Kurt Debattista, and Jungong Han. Dynamic contrastive learning guided by class confidence and confusion degree for medical image segmentation. *PR*, 2024. 2

[7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 2, 7

[8] Mu Chen, Zhedong Zheng, Yi Yang, and Tat-Seng Chua. Pipa: Pixel-and patch-wise self-supervised learning for domain adaptative semantic segmentation. In *ACM MM*, 2023. 2, 5, 6, 7

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2

[10] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 2

[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 *CVPR*, 2016. 1, 5

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 4, 1, 3

[13] 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. In *ICLR*, 2021. 2, 4

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5, 6, 7, 8

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 2

[16] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 2, 3, 4, 5, 6, 7, 1

[17] Pedro Hermosilla, Christian Stippel, and Leon Sick. Masked scene modeling: Narrowing the gap between supervised and self-supervised learning in 3d scene understanding. In *CVPR*, 2025. 2

[18] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *ICML*, 2018. 2

[19] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, 2022. 2, 3, 4, 5, 6, 7

[20] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. Mic: Masked image consistency for context-enhanced domain adaptation. In *CVPR*, 2023. 5, 6, 7, 2

[21] Lukas Hoyer, Dengxin Dai, Qin Wang, Yuhua Chen, and Luc Van Gool. Improving semi-supervised and domain-adaptive semantic segmentation with self-supervised depth estimation. *IJCV*, 2023. 2

[22] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Domain adaptive and generalizable network architectures and training strategies for semantic image segmentation. *TPAMI*, 2024. 2, 5, 6

[23] Tommie Kerssies, Niccolo Cavagnero, Alexander Hermans, Narges Norouzi, Giuseppe Averta, Bastian Leibe, Gijs Dubbelman, and Daan de Geus. Your vit is secretly an image segmentation model. In *CVPR*, 2025. 2

[24] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy and A. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2017. 2, 3, 5, 6, 7, 8

[25] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICMLW*, 2013. 3, 1

[26] Geon Lee, Chanho Eom, Wonkyung Lee, Hyekang Park, and Bumsub Ham. Bi-directional contrastive learning for domain adaptive semantic segmentation. In *ECCV*, 2022. 2

[27] Tianyu Li, Subhankar Roy, Huayi Zhou, Hongtao Lu, and Stéphane Lathuilière. Contrast, stylize and adapt: Unsupervised contrastive learning framework for domain adaptive semantic segmentation. In *CVPR*, 2023. 2

[28] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *ECCV*, 2022. 4

[29] Yinghong Liao, Wending Zhou, Xu Yan, Zhen Li, Yizhou Yu, and Shuguang Cui. Geometry-aware network for domain adaptive semantic segmentation. In *AAAI*, 2024. 5, 6

[30] Jinhong Lin, Cheng-En Wu, Huanran Li, Jifan Zhang, Yu Hen Hu, and Pedro Morgado. From prototypes to general distributions: An efficient curriculum for masked image modeling. In *CVPR*, 2025. 2

[31] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 2

[32] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *TPAMI*, 2021. 1

[33] Zhenliang Ni, Xinghao Chen, Yingjie Zhai, Yehui Tang, and Yunhe Wang. Context-guided spatial feature reconstruction for efficient semantic segmentation. In *ECCV*, 2024. 1

[34] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 2

[35] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 2

[36] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016. 5

[37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015. 2

[38] 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 *CVPR*, 2016. 5

[39] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *ICCV*, 2021. 1

[40] Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025. 2

[41] Changki Sung, Wanhee Kim, Jungho An, Wooju Lee, Hyungtae Lim, and Hyun Myung. Contextrast: Contextual contrastive learning for semantic segmentation. In *CVPR*, 2024. 1

[42] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000. 3

[43] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *WACV*, 2021. 2, 5, 6, 7, 1

[44] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *CVPR*, 2018. 2

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 4

[46] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *ICML*, 2019. 3

[47] Tung-Long Vuong, Hoang Phan, Vy Vo, Anh Bui, Thanh-Toan Do, Trung Le, and Dinh Phung. Preserving clusters in prompt learning for unsupervised domain adaptation. In *CVPR*, 2025. 2

[48] Haoqing Wang, Yehui Tang, Yunhe Wang, Jianyuan Guo, Zhi-Hong Deng, and Kai Han. Masked image modeling with local multi-scale reconstruction. In *CVPR*, 2023. 2, 3, 4

[49] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 2018. 1

[50] Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In *CVPR*, 2023. 2, 3, 5

[51] Binhui Xie, Shuang Li, Mingjia Li, Chi Harold Liu, Gao Huang, and Guoren Wang. Sepico: Semantic-guided pixel contrast for domain adaptive semantic segmentation. *TPAMI*, 2023. 2, 7

[52] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021. 2, 5, 7

[53] Linyan Yang, Lukas Hoyer, Mark Weber, Tobias Fischer, Dengxin Dai, Laura Leal-Taixé, Marc Pollefeys, Daniel Cremers, and Luc Van Gool. Micdrop: masking image and depth features via complementary dropout for domain-adaptive semantic segmentation. In *ECCV*, 2024. 2

[54] Yifei Zhang, Chang Liu, Jin Wei, Xiaomeng Yang, Yu Zhou, Can Ma, and Xiangyang Ji. Linguistics-aware masked image modeling for self-supervised scene text recognition. In *CVPR*, 2025. 2

[55] Zhe Zhang, Gaochang Wu, Jing Zhang, Xiatian Zhu, Dacheng Tao, and Tianyou Chai. Unified domain adaptive semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, 2025. 5, 6

[56] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao

Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*, 2021. 2

[57] Wenlve Zhou and Zhiheng Zhou. Unsupervised domain adaption harnessing vision-language pre-training. *TCSVT*, 2024. 1

[58] Yanfeng Zhou, Lingrui Li, Le Lu, and Minfeng Xu. nnwnet: Rethinking the use of transformers in biomedical image segmentation and calling for a unified evaluation benchmark. In *CVPR*, 2025. 2

[59] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018. 2