Unsupervised Domain Adaptation for Semantic Segmentation Based on Instance Spatial Geometry

Binglin Hao
School of Computer Science
Wuhan University
binglinhao@whu.edu.cn

Huajun Liu *
School of Computer Science
Wuhan University
huajunliu@whu.edu.cn

Abstract

Unsupervised domain adaptation (UDA) for semantic segmentation transfers knowledge from synthetic to real domains, where geometric cues such as depth are commonly exploited to reduce the domain gap. However, existing depth-aware methods do not explicitly model spatial relations among semantic instances that remain relatively stable across domains, while common self-training schemes rely primarily on prediction confidence to weight pseudo-labels without explicitly assessing geometric consistency. To address these issues, we propose an instance-wise geometric framework capturing inter-instance spatial relations, including directional concentration and dominant direction, beyond direct depth representations. These geometry-aware features are further integrated into pseudo-label weighting to promote geometric-semantic consistency during self-training, providing an explicit criterion for weighting target supervision. Experiments on standard benchmarks show our method significantly outperforms state-of-the-art UDA approaches. Code is available at https://github.com/binglinhao02/InstanceSpatialGeometry.

1 Introduction

Semantic segmentation aims to assign a semantic label to each pixel in an image. However, acquiring large-scale pixel-level annotations for real-world scenes is expensive and labor-intensive. Unsupervised domain adaptation (UDA) addresses this challenge by transferring models trained on labeled synthetic data (source domain) to unlabeled real-world data (target domain). Recent UDA methods for semantic segmentation, including DAFormer [7], HRDA [8], and MIC [9], have demonstrated improved performance across domains by combining advanced network architectures with effective adaptation strategies to reduce the discrepancy between synthetic and real images.

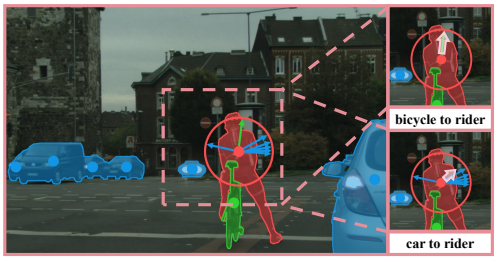

Figure 1: Spatial geometry captures directional distribution relative to a target instance: uniformly distributed supporting instances yield low directional concentration, whereas consistently aligned instances yield high directional concentration.

Depth information has been widely used to improve semantic segmentation and promote semantic consistency across domains. Many UDA methods exploit depth as an auxiliary geometric cue to facilitate adaptation. For example, SPIGAN [13] and DADA [28] introduce depth estimation as an auxiliary task, while CTRL [22] models the relationship

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

between semantic categories and depth levels within UDA. CorDA [30] employs self-supervised depth estimation to generate depth maps and adjusts pseudo-label weights during self-training based on depth discrepancies between source and target domains. Similarly, Wu and Liu [32] impose additional geometric constraints using depth distribution statistics. These approaches generally adopt multi-task learning that jointly optimizes semantic segmentation and depth estimation. MICDrop [34] further integrates semantic and depth modalities to improve their joint representation. Despite their effectiveness, these methods mainly rely on depth as the primary geometric representation and often overlook relatively stable spatial configurations among semantic instances across domains.

In UDA, self-training with pseudo-labels is a widely used strategy, where pseudo-labels are typically weighted by prediction confidence [25, 30, 7–9]. These confidence-based weights modulate pseudo-label supervision during target-domain learning. However, MetaCorrection [5] shows that prediction confidence is not always aligned with label correctness and proposes a noise transition matrix to address pseudo-label noise. Despite this observation, most existing methods rely primarily on prediction confidence for pseudo-label weighting, without explicitly assessing consistency with inter-instance geometry. We introduce geometric agreement as a complementary reliability cue.

The spatial arrangement of semantic instances exhibits relatively stable patterns across domains. As illustrated in Fig. 1, when cars are uniformly distributed around a target rider, the supporting directions exhibit high variance, resulting in low directional concentration. In contrast, when bicycles lie predominantly in similar directions relative to the rider, their supporting directions yield high directional concentration. However, such macroscopic geometric patterns are not explicitly assessed by pixel-wise semantic confidence alone. Thus, we model spatial relations at the instance level by characterizing the dominant direction and directional concentration. These relations are then broadcast to the feature map, enabling intermediate representations to encode global spatial structure. Consequently, pixel-level predictions are guided to remain consistent with these geometric patterns, providing robust structural cues under domain shift.

To this end, we introduce a metric to quantify inter-instance dominant direction and directional concentration in the joint image-depth space, resulting in a unified four-channel geometric representation composed of three direction channels and one concentration channel. Estimating this dense representation serves as an auxiliary task in our multi-task framework, acting as a structured alternative to direct depth supervision for more effective cross-domain transfer. Across both domains, we leverage correlations between the main and auxiliary tasks via a segmentation-geometry association loss. Utilizing geometric cues derived from segmentation labels, we promote cross-task consistency and stable convergence, especially on the target domain. Furthermore, two separate auxiliary semantic segmentation networks pre-trained with dominant direction and concentration collaborate with the main segmentation network to adjust pseudo-label weights, enhancing self-training robustness. Integrated with DAFormer, HRDA, and MIC, our method achieves consistent mIoU gains of 1.6, 1.9, and 1.1 on Synthia→Cityscapes, and 1.8, 1.4, and 1.3 on GTA→Cityscapes, respectively.

The main technical contributions of our work are summarized as follows:

- We introduce a novel method to quantify the spatial geometric distribution of semantic instances, serving as an effective bridge for cross-domain transfer in semantic segmentation. Rather than relying directly on depth information, our method models relative directional relationships between semantic instances, resulting in a clearer and more robust representation for domain adaptation.
- Within a multi-task framework, we introduce a segmentation-geometry association loss to explicitly model the relationship between the main segmentation and auxiliary spatial-geometric tasks. For the target domain, pseudo-geometric information can be used to regularize and enhance training.
- We propose a geometry-guided, pixel-wise pseudo-label reweighting strategy for self-training. Agreement between the semantic predictions of direction- and concentration-based auxiliary networks provides a geometric criterion for adjusting pseudo-label weights.

2 Related Work

Unsupervised Domain Adaptation. A common approach in UDA is to adapt a semantic segmentation model trained on synthetic datasets for deployment on real-world data. Most existing methods aim to reduce the domain gap through statistical distance minimization [14, 29], adversarial training [4, 29, 6, 15], or self-training [38, 40, 37, 19, 36]. Although these techniques have achieved notable

improvements by refining network architectures and training objectives, they often overlook the rich structural and geometric information embedded in both source and target domains.

Use of Geometric Information in Semantic Segmentation within a Multi-task Learning Framework. Most UDA methods exploit geometric cues via a multi-task learning framework, where depth estimation serves as an auxiliary task to support segmentation, as in SPIGAN [13], DADA [28], CTRL [22], CorDA [30], and Wu and Liu [32]. These approaches demonstrate that depth provides a geometric prior for reducing the domain gap. In contrast, MICDrop [34] leverages depth without adopting a multi-task formulation. However, existing depth-aware methods mainly rely on pixel-wise depth features, which capture only local geometry and ignore inter-instance spatial structures. Although auxiliary modalities such as text have been explored [17, 27], instance spatial distributions remain underexploited, motivating multi-task designs based on inter-instance relations.

Pseudo-label Reweighting in Self-training. Various strategies have been proposed to improve the reliability of pseudo-labels in self-training, including confidence thresholding [38, 40, 18], prototype-based labeling [37, 19, 36], and domain mixup [25, 7, 8, 39, 10]. However, MetaCorrection [5] shows that high prediction confidence does not necessarily correspond to semantic correctness at the pixel level. To address this limitation, we propose a pixel-wise pseudo-label reweighting strategy guided by geometric agreement, measured using inter-instance relative direction and directional concentration, to improve the stability and effectiveness of self-training.

3 Methodology

Our method exploits class-wise spatial association features among semantic instances in the joint image-depth coordinate space (x, y, z) to build a cross-domain transfer bridge between the source and target domains. This geometry-based prior facilitates domain adaptation and improves semantic segmentation performance on the target domain. An overview of the framework is shown in Fig. 2.

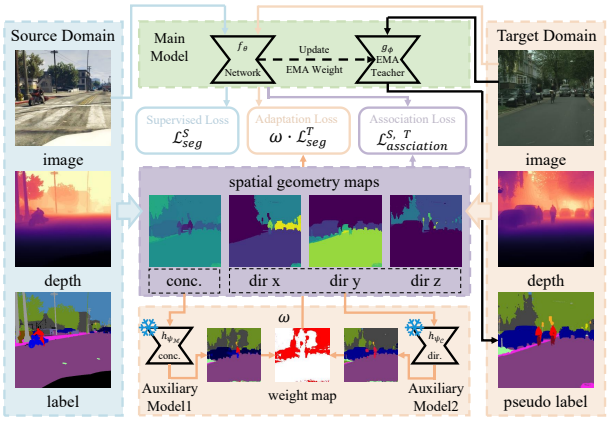

Figure 2: Method overview. Our framework extracts class-wise geometric cues (directional concentration and dominant direction) from semantic instances based on segmentation labels. These cues are incorporated as auxiliary supervision via an association loss to enhance the main segmentation network. Moreover, two separate auxiliary segmentation models are pre-trained on the source domain using these geometric cues, and their predictions are jointly exploited to refine pseudo-label weights during self-training.

Problem Definition. In UDA, we are given a labeled source domain dataset $(\mathbf{X}^S, \mathbf{Y}^S)$ and an unlabeled target domain dataset $\mathbf{X}^T$. The objective is to learn a segmentation model that generalizes to the target domain by reducing the discrepancy between $\mathbf{X}^S$ and $\mathbf{X}^T$. Performance is evaluated on a labeled validation set from the target domain.

Preliminaries. Source-domain training follows supervised learning. For UDA, we adopt a student-teacher framework [25, 1, 7–9, 26, 33], where the teacher is updated as an exponential moving average (EMA) [24] of the student and generates pseudo-labels for target images. These pseudo-labeled pairs $(\mathbf{X}^T, \hat{\mathbf{Y}}^T)$ are used to supervise the student. To ensure consistent geometric information across domains, depth maps are estimated for source and target images using Depth Anything V2 [35]. Ground-truth labels in the source domain and pseudo-labels in the target domain are employed to extract class-wise spatial geometry features for the auxiliary geometric task.

3.1 Spatial Geometry of Class Instances

In semantic segmentation, spatial geometry characterizes how instances of a semantic class are distributed relative to a target instance. Inspired by vector-based spatial association on a two-

dimensional plane [16], we extend this concept to the joint image-depth space. Within this space, to prevent dimensional dominance between image coordinates (x, y) and depth (z), we combine fixed-size crops with depth normalization ($z \in [0, 255]$). Furthermore, instead of enforcing physical isotropy in an inherently anisotropic space, we use the same coordinate scaling convention in both domains. Computing spatial geometry in this aligned space yields a pixel-wise feature map of directional concentration and dominant direction, which serves as a robust geometric alternative to direct depth for cross-domain semantic segmentation.

We first apply connected-component labeling to extract connected semantic components [31, 18, 11], which serve as approximate instances. For any pair of instances I_k^c and $I_j^{c'}$, where c and c' denote semantic classes and $k = 1, \ldots, N$, $j = 1, \ldots, M$ index instances within each class, we proceed as follows. We compute the centroids $\boldsymbol{\mu}_k^c$ and $\boldsymbol{\mu}_j^{c'}$ in the joint image-depth space. An association vector is then defined as $\mathbf{v}_{k,j}^{c,c'} = \boldsymbol{\mu}_k^c - \boldsymbol{\mu}_j^{c'}$, which is normalized to obtain the unit direction vector $\hat{\mathbf{v}}_{k,j}^{c,c'}$, representing the directional influence from $I_j^{c'}$ to I_k^c. To characterize the aggregate directional influence of all M instances of class c' on instance I_k^c, we compute the mean unit direction vector:

$$R_k^{c,c'} = \frac{1}{M} \sum_{j=1}^{M} \hat{\mathbf{v}}_{k,j}^{c,c'} = \left(x_k^{c,c'}, y_k^{c,c'}, z_k^{c,c'} \right) \tag{1}$$

Based on $R_k^{c,c'}$, we derive four scalar descriptors to represent its directional statistics. Specifically, a concentration measure $\mathcal{M}_k^{c,c'}$ is used to quantify the coherence of the underlying directional distribution, while three directional components $\mathcal{C}_{k,\cdot}^{c,c'}$ correspond to the dominant spatial orientation along the x, y, and z axes. They are defined as:

$$\begin{aligned} \mathcal{M}_k^{c,c'} &= \sqrt{\left(x_k^{c,c'}\right)^2 + \left(y_k^{c,c'}\right)^2 + \left(z_k^{c,c'}\right)^2}, \\ \mathcal{C}_{k,x}^{c,c'} &= \frac{x_k^{c,c'}}{\mathcal{M}_k^{c,c'}}, \quad \mathcal{C}_{k,y}^{c,c'} = \frac{y_k^{c,c'}}{\mathcal{M}_k^{c,c'}}, \quad \mathcal{C}_{k,z}^{c,c'} = \frac{z_k^{c,c'}}{\mathcal{M}_k^{c,c'}} \end{aligned} \tag{2}$$

The concentration term $\mathcal{M}_k^{c,c'}$ measures the coherence of the directional vectors, where larger values indicate strong directional consistency and smaller values correspond to dispersed configurations with mutual cancellation. The normalized components $\mathcal{C}_{k,\cdot}^{c,c'}$ represent the dominant geometric orientation by projecting the resultant direction onto the coordinate axes, thereby explicitly encoding the relative direction from supporting instances to the target instance. Boundary cases are specified in Sec. B.

To model the global geometric context of an instance I_k^c, we aggregate its spatial geometric statistics across all semantic classes. The class-wise descriptors are averaged to form a spatial association representation that summarizes the geometric relations between the instance and the scene:

$$\left(\mathcal{M}_k^c, \mathcal{C}_{k,x}^c, \mathcal{C}_{k,y}^c, \mathcal{C}_{k,z}^c\right) = \left(\frac{1}{C} \sum_{c'=1}^{C} \left[\mathcal{M}_k^{c,c'}, \mathcal{C}_{k,x}^{c,c'}, \mathcal{C}_{k,y}^{c,c'}, \mathcal{C}_{k,z}^{c,c'} \right] \right) \tag{3}$$

The resulting four-channel geometric descriptor is propagated back to all pixels of instance I_k^c. Applying this process to all instances produces a dense geometric feature map $\mathbf{S}$ for the entire image. This top-down broadcasting bridges instance-wise and pixel-wise representations. The map $\mathbf{S}$ is then used as the supervisory signal for the association loss in both the source and target domains. Through this loss, the dense geometric representation imposes structural constraints that encourage pixel-wise predictions to reflect instance-wise geometric relations, as detailed in Sec. 3.2.

3.2 Segmentation-Geometry Association Loss

In the multi-task learning framework, the main segmentation branch is optimized using the standard cross-entropy loss. Ground-truth labels $\mathbf{y}^S$ are adopted for the source domain, while pseudo-labels $\hat{\mathbf{y}}^T$ are used for the target domain. For target-domain training, we follow a self-training paradigm with a pixel-wise weighting factor ω to alleviate the impact of noisy pseudo-labels:

$$\mathcal{L}_{seg}^{S}\left(\mathbf{X}^{S},\mathbf{Y}^{S}\right)=\mathbb{E}\left[-\mathbf{y}^{S}\log\overline{\mathbf{y}}_{seg}^{S}\right] \tag{4}$$

$$\mathcal{L}_{seg}^{T}\left(\mathbf{X}^{T},\hat{\mathbf{Y}}^{T}\right)=\mathbb{E}\left[-\omega\hat{\mathbf{y}}^{T}\log\overline{\mathbf{y}}_{seg}^{T}\right] \tag{5}$$

where $\overline{\mathbf{y}}_{seg}$ denotes the predicted semantic segmentation map.

To impose structural constraints on the auxiliary geometric branch, we introduce the **Segmentation-Geometry Association Loss**. In the source domain, geometric targets $\mathbf{S}^{S}$ are derived from ground-truth labels. In the target domain, geometric targets $\hat{\mathbf{S}}^{T}$ are constructed from teacher pseudo-labels $\hat{\mathbf{y}}^{T}$ and estimated depth via the same deterministic mapping (Eqs. (1) to (3)). Because these targets may carry teacher pseudo-label errors, target-domain association supervision is activated only after the initial training stage to limit the influence of early noise. The geometric branch is optimized in both domains using the reverse Huber (BerHu) loss [12]:

$$\mathcal{L}_{association}^{S}\left(\mathbf{X}^{S},\mathbf{S}^{S}\right)=\mathbb{E}\left[\mathrm{berHu}\left(\overline{\mathbf{S}}^{S}-\mathbf{S}^{S}\right)\right] \tag{6}$$

$$\mathcal{L}_{association}^{T}\left(\mathbf{X}^{T},\hat{\mathbf{S}}^{T}\right)=\mathbb{E}\left[\mathrm{berHu}\left(\overline{\mathbf{S}}^{T}-\hat{\mathbf{S}}^{T}\right)\right] \tag{7}$$

where $\overline{\mathbf{S}}$ denotes the predicted geometric feature maps. The BerHu loss is defined as:

$$\begin{gathered}\mathrm{berHu}\left(e_{S}\right)=\begin{cases}\left|e_{S}\right|, & \left|e_{S}\right|\leq H\\ \frac{\left(e_{S}\right)^{2}+H^{2}}{2H}, & \left|e_{S}\right|>H\end{cases}\\ H=0.2\max\left(\left|e_{S}\right|\right)\end{gathered} \tag{8}$$

The four-channel residual e_S is computed element-wise over pixels i and channels c. Minimizing this residual couples semantic and geometric representations through shared features. In the unlabeled target domain, self-training supervises the semantic branch with pseudo-labels, whereas the auxiliary geometric branch would otherwise lack direct target-domain supervision. The association loss supplies an inter-instance relational objective for this branch. As shown in Table 2, adding target-domain association improves segmentation performance.

We incorporate spatial geometric features as an auxiliary multi-task signal. Following the feature fusion strategy of CorDA [30], we replace domain-specific depth decoders with a shared geometric decoder to enable joint optimization across both domains.

The overall training objective is defined as

$$\begin{aligned}\mathcal{L}=&\tilde{\mathcal{L}}_{seg}^{S}+\tilde{\mathcal{L}}_{seg}^{T}+\lambda\,\tilde{\mathcal{L}}_{association}^{S}+\lambda\,\tilde{\mathcal{L}}_{association}^{T}\\&+\mathcal{L}_{seg}^{S}+\mathcal{L}_{seg}^{T}+\lambda\,\mathcal{L}_{association}^{S}+\lambda\,\mathcal{L}_{association}^{T}\end{aligned} \tag{9}$$

where λ balances the contribution of the geometric association loss and is empirically set to $\lambda=0.5$. The tilde in $\tilde{\mathcal{L}}$ denotes loss terms applied to intermediate predictions, following the design in CorDA. For HRDA and MIC, these intermediate losses, including $\tilde{\mathcal{L}}_{\text{seg}}^{S}$, $\tilde{\mathcal{L}}_{\text{seg}}^{T}$, $\tilde{\mathcal{L}}_{\text{association}}^{S}$, and $\tilde{\mathcal{L}}_{\text{association}}^{T}$, are applied only to the low-resolution context branch.

3.3 Pseudo-label Weight Adjustment Guided by Geometric Cues

Pseudo-label reliability in UDA is often affected by domain-induced ambiguities. To address this issue, we jointly exploit two complementary geometric cues: dominant direction, which characterizes the dominant geometric direction of a semantic class, and directional concentration, which measures how tightly those directions are clustered. Their interaction provides a criterion for identifying geometrically consistent pseudo-labels.

Method	Depth	Road	Side.	Buil.	Wall	Fence	Pole	Light	Sign	Vege.	Terr.	Sky	Pers.	Rider	Car	Truck	Bus	Train	Mbike	Bike	mIoU
Synthia→Cityscapes (Val.)																					
CTRL [22]	✓	86.9	43.0	80.7	19.2	0.9	27.2	11.6	12.6	81.3	-	83.2	60.7	24.0	84.2	-	46.2	-	22.0	44.2	45.5
CorDA [30]	✓	**93.3**	**61.6**	85.3	19.6	5.1	37.8	36.6	42.8	84.9	-	90.4	69.7	41.8	85.6	-	38.4	-	32.6	53.9	55.0
Wu and Liu [32]	✓	85.3	40.2	79.7	19.6	1.3	29.4	29.7	32.2	82.5	-	79.2	64.3	26.7	85.2	-	49.4	-	22.7	44.9	48.2
CoPT [17]	-	83.4	44.3	90.0	50.4	8.0	60.0	67.0	63.0	87.5	-	94.8	81.1	58.6	89.7	-	66.5	-	**68.9**	65.0	67.4
DAFormer [7]	-	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/ MICDrop [34]	✓	81.0	37.1	89.4	45.7	9.5	51.8	57.3	58.0	86.7	-	85.0	73.6	50.4	88.2	-	64.7	-	56.8	62.8	62.4
w/ Ours	✓	87.8	48.2	88.4	37.8	9.0	50.6	55.2	53.3	87.6	-	90.6	73.7	48.0	87.6	-	63.3	-	55.8	62.6	62.5
HRDA [8]	-	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/ MICDrop [34]	✓	86.3	49.6	89.3	**53.7**	5.1	57.6	66.4	63.8	86.1	-	94.1	79.1	56.0	87.8	-	65.0	-	64.2	65.0	66.8
w/ Ours	✓	88.9	50.5	90.4	48.4	7.9	60.1	65.8	62.5	89.0	-	94.4	79.9	56.2	90.7	-	67.9	-	63.6	67.3	67.7
MIC [9]	-	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/ MICDrop [34]	✓	82.8	42.6	**90.5**	51.6	**9.6**	61.0	65.7	**65.0**	89.1	-	**95.0**	81.1	59.7	90.6	-	68.3	-	67.4	66.5	67.9
w/ Ours	✓	91.2	55.8	88.7	39.7	**9.6**	**62.1**	**69.0**	55.0	**89.9**	-	94.3	**81.5**	**60.4**	**91.2**	-	**71.1**	-	67.3	**67.7**	**68.4**
GTA→Cityscapes (Val.)																					
CorDA [30]	✓	94.7	63.1	87.6	30.7	40.6	40.2	47.8	51.6	87.6	47.0	89.7	66.7	35.9	90.2	48.9	57.5	0.0	39.8	56.0	56.6
CoPT [17]	-	97.6	80.9	91.6	62.1	55.9	59.3	**66.7**	70.5	**91.9**	53.0	94.4	80.0	55.6	94.7	87.1	88.6	82.1	65.0	68.8	76.1
DAFormer [7]	-	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/ MICDrop [34]	✓	96.0	71.8	90.3	53.3	46.4	54.8	57.8	66.7	90.0	49.2	92.2	73.6	46.3	92.8	78.1	80.6	70.7	57.5	63.2	70.1
w/ Ours	✓	96.7	75.3	89.4	50.5	46.7	51.5	58.1	61.3	89.9	50.2	91.0	74.4	48.6	93.1	77.5	82.3	74.2	59.6	61.3	70.1
HRDA [8]	-	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/ MICDrop [34]	✓	95.8	71.1	91.5	**62.8**	55.0	60.8	64.0	73.4	91.3	49.1	94.0	79.2	54.6	94.4	84.8	88.5	79.0	65.9	65.5	74.8
w/ Ours	✓	96.6	75.3	91.4	62.3	56.4	60.6	66.1	72.4	91.1	49.2	93.4	79.1	54.8	94.1	82.2	90.4	**82.5**	65.0	64.9	75.2
MIC [9]	-	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/ MICDrop [34]	✓	97.6	81.5	**92.0**	**62.8**	59.4	**62.6**	62.9	**73.6**	91.6	52.6	94.1	80.2	57.0	94.8	**87.4**	**90.7**	81.6	65.3	67.8	76.6
w/ Ours	✓	**97.8**	**82.9**	91.9	61.2	**60.3**	62.2	65.5	72.6	91.8	**54.0**	**94.5**	**81.7**	**59.7**	**94.9**	86.6	88.9	82.4	**66.2**	**71.8**	**77.2**

Table 1: Comparison with state-of-the-art UDA methods across benchmarks in IoU (%). The "w/" rows within each block denote separate additions to that framework (DAFormer, HRDA, or MIC).

Based on these cues, we pre-train two auxiliary semantic segmentation networks on the source domain, each specialized in one geometric cue. By decoupling dominant direction and directional concentration into separate streams, the two networks provide complementary assessments of geometric consistency, used to guide pixel-wise pseudo-label weight adjustment during adaptation.

Both auxiliary networks share the same architecture, consisting of a SegFormer-based encoder and a DAFormer decoder, and predict in the same semantic space as the main network. They are pre-trained solely on source-domain data and frozen during self-training, serving as geometry-aware anchors between the source and target domains.

During adaptation, pixel-wise predictions from the two auxiliary networks are compared to adjust pseudo-label weights. If the predictions agree, the pseudo-label is treated as more reliable and its training weight is amplified by a factor α ($\alpha > 1$). Otherwise, the weight is attenuated by a factor β ($0 < \beta < 1$), reducing the influence of ambiguous supervision:

$$\omega' = \begin{cases} \alpha \cdot \omega, & \hat{\mathbf{y}}^T_{conc} = \hat{\mathbf{y}}^T_{dir} \\ \beta \cdot \omega, & \hat{\mathbf{y}}^T_{conc} \neq \hat{\mathbf{y}}^T_{dir} \end{cases} \tag{10}$$

where ω' denotes the refined pixel-wise weight applied to target-domain pseudo-labels in both $\tilde{\mathcal{L}}^T_{\text{seg}}$ and $\mathcal{L}^T_{\text{seg}}$ (see Eq. (9)). These hyperparameters are kept fixed at $\alpha = 1.1$ and $\beta = 0.8$ throughout all experiments on both GTA→Cityscapes and Synthia→Cityscapes. A detailed sensitivity analysis is provided in Sec. 4.4.

4 Experiments and Analysis

Datasets and Metrics. Following prior work [30, 7–9, 34], we conduct experiments on two standard benchmarks: GTA→Cityscapes and Synthia→Cityscapes. The source domains include GTA [20] (24,966 images, 1914×1052) and Synthia [21] (9,400 images, 1280×760), while the target domain Cityscapes [3] provides 2,975 training and 500 validation images (2048×1024). Regarding input resolution, for low-resolution settings, both Cityscapes and GTA are resized to 1024×512; for high-resolution settings, Cityscapes maintains its original resolution, while GTA and Synthia are resized to 2048×1024 and 2560×1520. We report results using mean Intersection over Union

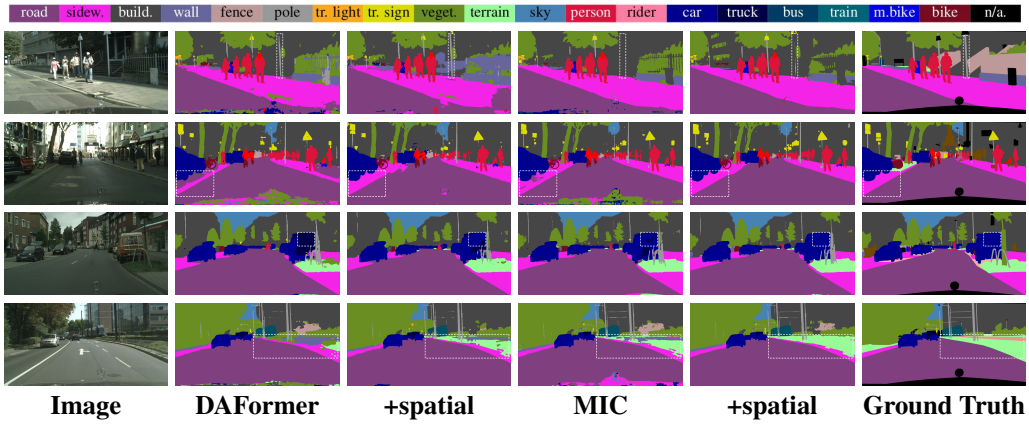

Figure 3: Qualitative results for Synthia→Cityscapes (top two rows) and GTA→Cityscapes (bottom two rows). Compared to DAFormer and MIC, our method improves predictions for *pole*, *sidewalk*, *car* and *terrain*, with enhancements highlighted in white boxes.

(mIoU) as a percentage for 19 common categories on GTA→Cityscapes, and 16 common categories on Synthia→Cityscapes.

Network Architecture. We adopt DAFormer as the primary baseline for component analysis, owing to its trade-off between performance and efficiency. To demonstrate the generality of our approach, we further integrate it with other state-of-the-art methods, including HRDA and MIC.

Training Details. Our method is integrated into DAFormer, HRDA, and MIC, retaining their original training setups, including the optimizer, warm-up strategy, batch size, learning rate, and EMA scheme. All models are trained for 60k iterations on an NVIDIA A100 GPU.

4.1 Main Results

As summarized in Table 1, integrating our method consistently improves mIoU across all baseline frameworks. This validates the general effectiveness of exploiting dominant direction and directional concentration as structure-aware geometric cues defined over inter-instance relations. Despite this consistency, the magnitude and distribution of gains exhibit distinct dataset-dependent patterns.

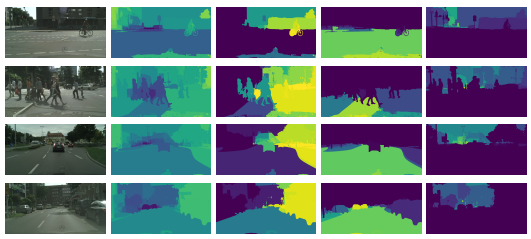

Figure 4: Auxiliary supervision signals. From left to right: the image, the directional concentration, and the x, y, z dominant direction components.

Notably, on the Synthia→Cityscapes benchmark, our method achieves consistent improvements on categories with stable spatial layouts, such as *road* (**+4.6**) and *sidewalk* (**+5.3**), while also reducing confusion in complex object categories, such as *bus* (**+9.2**) and *bicycle* (**+3.4**). For GTA→Cityscapes, the gains mainly arise from improved fine-grained geometric consistency. This is evidenced by better performance on structurally delicate categories such as *fence* (**+3.4**) and *pole* (**+2.5**), and improved performance for dynamic categories like *rider* and *bicycle* (**>+2.0**). Overall, these results suggest our method consistently enhances both global layout understanding and local structure prediction across different domain gaps.

Furthermore, we compare our method against the SOTA depth-based approach, MICDrop. As evidenced by the results, our approach achieves superior performance in five of six experimental settings, while maintaining parity on the remaining one. This confirms that our geometric cues yield robust and consistent improvements across diverse frameworks, enhancing cross-domain semantic segmentation performance.

Fig. 3 compares our qualitative results against DAFormer and MIC. Our method yields smoother and more consistent predictions for global structures (e.g., *sidewalk*, *terrain*) while simultaneously capturing finer details in boundary-sensitive classes like *pole* and *car*. This further validates the

Method	Association Loss		Pseudo Label Adjustment	mIoU
	source	target		
M1	-	-	-	68.3 ±0.5
M2	✓	-	-	68.6 ±0.4
M3	✓	✓	-	69.2 ±0.3
M4	-	-	✓	69.2 ±0.6
M5(2D)	✓	✓	✓	69.6 ±0.5
M6(3D)	✓	✓	✓	70.1 ±0.3

Table 2: Ablation of proposed components using DAFormer on GTA→Cityscapes. Results report the mean ± std over 3 runs.

Method	Auxiliary type	Auxiliary detail	mIoU
M7	depth	depth	68.8 ±0.3
M8	structure	concentration only	68.7 ±0.6
M9	structure	direction only	69.0 ±0.4
M3	structure	conc. + dir.	69.2 ±0.3

Table 3: Ablation of different auxiliary tasks using DAFormer on GTA→Cityscapes. Results report the mean ± std over 3 runs.

advantage of incorporating geometric priors to refine domain adaptation. Additional visualizations, including results for HRDA, are available in the supplementary material. Visualizations of the proposed geometric features are provided in Fig. 4.

4.2 Ablation Studies for Different Components

We conduct comprehensive ablation studies on the GTA→Cityscapes benchmark using the DAFormer baseline. First, Table 2 disentangles the contributions of our proposed components. Starting from the DAFormer baseline (M1), applying the association loss only on the source domain (M2) yields consistent improvements, indicating that geometric association provides complementary supervision. Extending this loss to the target domain (M3) further improves performance by enforcing cross-branch regularization, promoting balanced optimization. Similar to pseudo-label self-training, error accumulation from noisy pseudo-labels may still occur; the association loss nevertheless leads to improved target-domain performance of the main task. Using pseudo-label weighting alone (M4) yields comparable improvements, indicating its effectiveness in mitigating noise in target-domain supervision. Combining both components yields the best results, with M6 outperforming M5 by constructing geometric features in the joint image–depth (3D) space rather than in 2D image space. This highlights that richer geometric representations provide stronger structural guidance.

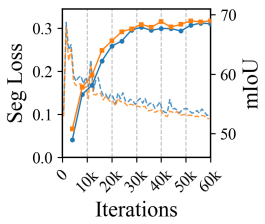

Figure 5: Training dynamics of M2 and M3.

To further elucidate the impact of cross-branch regularization, Fig. 5 plots the training dynamics for M2 (blue) and M3 (orange). As the curves indicate, after target-domain association supervision is activated, M3 generally achieves higher mIoU (solid lines) and maintains a smoother, slightly lower segmentation loss (dashed lines) than M2. Together, these trends support the benefit of target-domain association supervision despite the risk of error propagation from noisy pseudo-labels.

Furthermore, Table 3 analyzes geometric cues within this framework, disabling pseudo-label weight adjustment to strictly isolate the effects of auxiliary supervision. While depth-based supervision (M7) yields consistent gains by providing valuable spatial context, it cannot fully exploit geometric cues without our instance spatial modeling. Our individual structural cues also yield improvements, with direction (M9) proving more informative than concentration (M8). Combining both (M3) achieves the best performance, indicating their complementary nature. Overall, our geometric representation constitutes an effective relational alternative to direct depth supervision for cross-domain semantic segmentation.

4.3 Interpretability of Geometric Guidance

To illustrate how geometric priors guide pixel-level classification, we apply Grad-CAM [23, 2] to visualize gradient responses from the shared feature layer for structurally ambiguous categories (e.g., *pole* and *sidewalk*) under domain shift. As shown in Fig. 6, the DAFormer activations are concentrated mainly on pixels of the queried semantic class. While such texture-driven responses are necessary, they lack structural support and fail to capture contextual cues from surrounding classes, making predictions unstable under domain shift.

In contrast, our approach produces broader, structure-aware activations. In addition to strong intra-class responses, the model highlights inter-class boundaries and surrounding context. Guided by

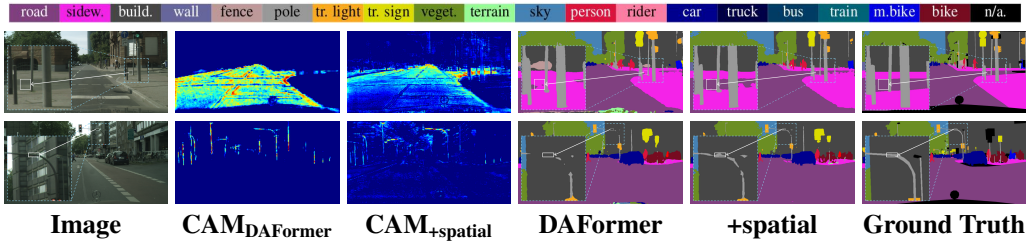

Figure 6: Grad-CAM interpretability analysis. The white boxes in the segmentation maps indicate the challenging pixels selected as the target for Grad-CAM generation.

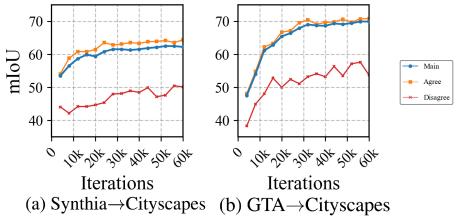

Figure 7: Comparison of mIoU evolution.

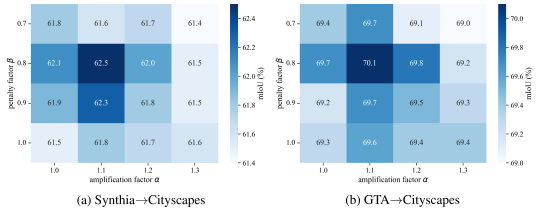

Figure 8: Hyperparameter sensitivity analysis.

auxiliary geometric supervision, these activations encode consistent spatial relationships that provide structural support for pixel-level predictions. As a result, predictions are no longer made in isolation but are informed by the global scene layout, leading to more reliable inference in ambiguous regions. Overall, this indicates that our method moves beyond appearance-based matching and instead grounds dense predictions in geometric structure.

4.4 Justification and Robustness of Pseudo-Label Adjustment

To evaluate the proposed geometric guidance, we analyze mIoU evolution conditioned on the agreement between two auxiliary predictions (Fig. 7). The training curves exhibit a clear performance stratification: regions with geometric agreement (orange) consistently achieve higher mIoU than the overall baseline (blue), whereas disagreement regions (red) show substantially lower performance. This divergence suggests that the proposed geometric cues capture complementary structural priors, where agreement correlates with semantic consistency and disagreement highlights ambiguous regions in the target domain.

	Agree	Disagree
Pixel share (%)	85.8	14.2
Correct rate (%)	94.7	85.4
Error rate (%)	5.3	14.7
Share of errors (%)	68.7	31.3

Table 4: Pseudo-label reliability by geometric agreement.

To examine whether this stratification is also reflected in pseudo-label quality, we analyze one DAFormer GTA→Cityscapes run over iterations 40k–60k. As shown in Table 4, agreement pixels have higher accuracy than disagreement pixels (94.7% versus 85.4%). Disagreement covers only 14.2% of pixels but contains 31.3% of all errors; its error rate is 14.7%, compared with 5.3% for agreement. Thus, geometric agreement separates pseudo-labels into groups with different reliability, supporting their differential weighting during training.

Guided by these observations, we increase pseudo-label weights in agreement regions and decrease them in disagreement regions. The hyperparameters are fixed at $\alpha = 1.1$ and $\beta = 0.8$ across all frameworks and benchmarks. As shown in Fig. 8, performance peaks at the selected setting, but every tested combination outperforms the DAFormer baseline. Thus, within the evaluated range, the parameters affect the magnitude of the gain rather than whether it occurs. Overall, the proposed strategy uses an explicit geometric-agreement signal alongside semantic confidence, delivering consistent gains without requiring extensive tuning.

4.5 Quantitative Cross-Domain Consistency

We quantify cross-domain consistency of geometric descriptors using GTA training and Cityscapes validation labels. For connected semantic instances larger than 50 pixels, we compute depth statistics

feat.	$W_1 \downarrow$	$\mathrm{MMD}^2 \downarrow$	BA$\rightarrow 0.5$	AUC$\rightarrow 0.5$
depth	0.471	0.0581	0.649	0.693
centroid	0.348	0.0356	0.669	0.724
2D desc.	**0.293**	**0.0236**	**0.587**	**0.607**
3D desc.	0.331	0.0320	0.611	0.649

Table 5: Cross-domain discrepancy of semantic-instance representations on GTA→Cityscapes.

Method	Target	mIoU
M1	None	68.3 ± 0.5
M10	Img. shuffled	68.1 ± 0.6
M11	100% Inst. shuffled	68.5 ± 0.7
M12	50% Inst. shuffled	68.6 ± 0.4
M13	25% Inst. shuffled	68.8 ± 0.5
M3	Correct	69.2 ± 0.3

Table 6: Effect of geometric shuffling with DAFormer on GTA→Cityscapes.

(depth), image-depth centroids (centroid), and 2D and 3D descriptors (2D desc. and 3D desc.). Per class, we sample up to ten instances per image, match domain counts, and repeat 20 times. Features are standardized using source-domain channel statistics. Class-conditional W_1 measures channel-wise marginal differences; MMD^2 measures joint differences using three RBF kernels with source-derived bandwidths. An MLP with image-grouped ten-fold validation yields balanced accuracy (BA) for domain classification at a threshold of 0.5 and threshold-independent ROC AUC.

Both geometric descriptors show lower discrepancy than depth statistics and image-depth centroids on all four metrics in Table 5. The 2D descriptor shows the lowest discrepancy; the 3D descriptor has greater discrepancy but remains below depth statistics and image-depth centroids. Image-depth centroids have lower W_1 and MMD^2 than depth statistics but higher MLP BA/AUC, showing that distributional distances and domain discriminability need not agree. Although the 3D descriptor has greater measured discrepancy than the 2D descriptor, it achieves higher segmentation performance, suggesting that depth contributes task-useful information. These results support relative cross-domain consistency in this setting.

4.6 Effect of Geometric Correspondence

We test whether the benefit of geometric supervision depends on meaningful instance-level spatial correspondence rather than merely adding an auxiliary loss. Image shuffling swaps complete four-channel geometric maps between images in a batch, preserving each map's internal structure while breaking its correspondence with the current scene. Instance shuffling reassigns intact four-dimensional descriptors among semantic class instances within an image, preserving instance masks and channel coupling while breaking descriptor-to-instance correspondence. The percentages indicate the fraction of instances whose descriptors are reassigned.

In Table 6, correct geometry increases mIoU from 68.3 to 69.2, whereas image shuffling reduces it to 68.1. Instance shuffling yields 68.5, 68.6, and 68.8 mIoU as the shuffled fraction decreases from 100% to 50% and 25%. This graded recovery, together with the 0.7 gap between fully shuffled and correct geometry, indicates that the descriptor retains useful spatial relations whose alignment with the image and semantic instance accounts for much of the observed gain.

5 Conclusions and Limitations

We introduce an instance-level geometric representation based on dominant direction and directional concentration to capture relatively stable inter-instance spatial structure for cross-domain semantic segmentation. By leveraging these geometric cues both as auxiliary supervision and for pseudo-label weight refinement, the proposed method consistently improves performance across benchmarks and frameworks, outperforming state-of-the-art depth-aware UDA approaches. The depth-enhanced 3D variant depends on estimated depth and may be sensitive to depth errors or cross-domain inconsistencies, while the depth-free 2D variant cannot capture relative depth. Both variants may also be affected by shifts in class-instance layouts. Future work will focus on improving robustness to depth errors and layout shifts.

Acknowledgments. This work is funded by the National Natural Science Foundation of China (No.41771427). We thank the anonymous reviewers for their valuable feedback and time.

References

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

[2] Zhaozheng Chen, Tan Wang, Xiongwei Wu, Xian-Sheng Hua, Hanwang Zhang, and Qianru Sun. Class re-activation maps for weakly-supervised semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 969–978, 2022.

[3] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016.

[4] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17:1–35, 2016.

[5] Xiaoqing Guo, Chen Yang, Baopu Li, and Yixuan Yuan. Metacorrection: Domain-aware meta loss correction for unsupervised domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3927–3936, 2021.

[6] 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 *Proceedings of the 35th International Conference on Machine Learning*, pages 1989–1998, 2018.

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

[8] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. HRDA: Context-aware high-resolution domain-adaptive semantic segmentation. In *European Conference on Computer Vision*, pages 372–391, 2022.

[9] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. MIC: Masked image consistency for context-enhanced domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11721–11732, 2023.

[10] 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. *International Journal of Computer Vision*, 131(8):2070–2096, 2023.

[11] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Cross-view regularization for domain adaptive panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10133–10144, 2021.

[12] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 239–248, 2016.

[13] Kuan-Hui Lee, Germán Ros, Jie Li, and Adrien Gaidon. SPIGAN: Privileged adversarial learning from simulation, 2018. Preprint.

[14] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I. Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 2208–2217. PMLR, 2017.

[15] Yawei Luo, Ping Liu, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Category-level adversarial adaptation for semantic segmentation using purified features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):3940–3956, 2021.

[16] Kanti V. Mardia and Peter E. Jupp. *Directional Statistics*. John Wiley & Sons, 2009.

[17] Cristina Mata, Kanchana Ranasinghe, and Michael S. Ryoo. Copt: Unsupervised domain adaptive segmentation using domain-agnostic text embeddings. In *European Conference on Computer Vision*, 2025.

[18] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. In *European Conference on Computer Vision*, pages 415–430, 2020.

[19] Yingwei Pan, Ting Yao, Yehao Li, Yu Wang, Chong-Wah Ngo, and Tao Mei. Transferrable prototypical networks for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2239–2247, 2019.

[20] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *European Conference on Computer Vision*, pages 102–118, 2016.

[21] Germán Ros, Laura Sellart, Joanna Materzynska, David Vázquez, and Antonio M. López. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3234–3243, 2016.

[22] Suman Saha, Anton Obukhov, Danda Pani Paudel, Menelaos Kanakis, Yuhua Chen, Stamatios Georgoulis, and Luc Van Gool. Learning to relate depth and semantics for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8197–8207, 2021.

[23] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[24] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, page 1195–1204, 2017.

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

[26] Thanh-Dat Truong, Ngan Le, Bhiksha Raj, Jackson Cothren, and Khoa Luu. FREDOM: Fairness domain adaptation approach to semantic scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19988–19997, 2023.

[27] Thanh-Dat Truong, Utsav Prabhu, Dongyi Wang, Bhiksha Raj, Susan Gauch, Jeyamkondan Subbiah, and Khoa Luu. Eagle: efficient adaptive geometry-based learning in cross-view understanding. In *Advances in Neural Information Processing Systems*, pages 137309–137333, 2024.

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

[29] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7028–7037, 2019.

[30] Qin Wang, Dengxin Dai, Lukas Hoyer, Luc Van Gool, and Olga Fink. Domain adaptive semantic segmentation with self-supervised depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8515–8525, 2021.

[31] Zhonghao Wang, Mo Yu, Yunchao Wei, Rogerio Feris, Jinjun Xiong, Wen-mei Hwu, Thomas S Huang, and Honghui Shi. Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12635–12644, 2020.

[32] Quanliang Wu and Huajun Liu. Unsupervised domain adaptation for semantic segmentation using depth distribution. In *Advances in Neural Information Processing Systems*, pages 14374–14387, 2022.

[33] Binhui Xie, Shuang Li, Mingjia Li, Chi Harold Liu, Gao Huang, and Guoren Wang. Sepico: Semantic-guided pixel contrast for domain adaptive semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):9004–9021, 2023.

[34] 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 *European Conference on Computer Vision*, pages 329–346, 2024.

[35] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. In *Advances in Neural Information Processing Systems*, pages 21875–21911, 2024.

[36] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12414–12424, 2021.

[37] Qiming Zhang, Jing Zhang, Wei Liu, and Dacheng Tao. Category anchor-guided unsupervised domain adaptation for semantic segmentation. In *Advances in Neural Information Processing Systems*, 2019.

[38] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3801–3809, 2018.

[39] Qianyu Zhou, Zhengyang Feng, Qiqi Gu, Jiangmiao Pang, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. Context-aware mixup for domain adaptive semantic segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(2):804–817, 2022.

[40] Yang Zou, Zhiding Yu, B. V. K. Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *European Conference on Computer Vision*, pages 289–305, 2018.

Appendix

Overview

The supplementary material provides additional details on the proposed method, along with extended experimental results and analyses. It includes descriptions of the instance extraction strategy (Sec. A) and boundary conditions for geometric feature computation (Sec. B), as well as the rationale for global geometric aggregation (Sec. C), scale consistency in the joint image-depth space (Sec. D), and source-domain pretraining of the auxiliary networks (Sec. E).

We further present analyses of learning dynamics for spatially structured classes (Sec. F) and computational efficiency (Sec. G), followed by t-SNE feature visualization (Sec. H) and component-wise ablations of the dominant direction cue (Sec. I). Qualitative comparisons across DAFormer, HRDA, and MIC (Sec. J) and a discussion of broader societal impacts (Sec. K) are also provided.

A Instance Extraction Strategy

To derive instance-level geometric information without instance-level annotations, we use connected-component labeling (CCL) to generate pseudo-instance masks from source ground-truth semantic labels and target pseudo-labels. We apply 8-neighbor connectivity to group adjacent pixels of the same semantic class into connected components, which serve as approximate object instances. This separation ensures that spatially disjoint regions receive unique IDs, facilitating the independent computation of geometric centers, dominant direction, and directional concentration statistics.

To enhance numerical stability and mitigate spurious signals from fragmented artifacts, we treat connected components containing fewer than K pixels as noise. We consistently set this threshold to $K = 50$ across all experiments. Furthermore, to ensure computational efficiency, the core instance extraction module is implemented in C++ and compiled as a dynamic shared object (.so) library. This optimized implementation integrates seamlessly into the Python-based training pipeline for on-the-fly mask generation, imposing minimal overhead on data loading and model optimization.

B Boundary Conditions for Geometric Feature Calculation

To ensure numerical stability, well-defined geometry, and consistent optimization behavior, we explicitly specify boundary handling protocols for the computation of the proposed direction and directional concentration maps. These protocols address degenerate cases that naturally arise in instance-level geometric modeling and are essential for robust large-scale training, serving as a prerequisite for the stable evaluation of Eqs. (1) to (3).

1. Handling Absent Categories. When a supporting category c' is absent from an image, its relations to a query instance I_k^c are undefined. We therefore set its class-wise concentration and three directional components to zero. These zero values remain included in the average over the fixed set of C semantic classes in Eq. (3), keeping the descriptor dimension and aggregation rule consistent across images.

2. Intra-Class Self-Exclusion. When computing instance-level geometric features, we explicitly handle degenerate self-relations. For a query instance I_k^c, including the instance itself would result in trivial zero displacement vectors. When the supporting category matches the query category ($c' = c$), we exclude the query instance and compute relations using only the remaining instances of that category. If an instance represents the sole occurrence of its class, the supporting set becomes empty. This scenario defaults to the zero-initialization strategy, maintaining consistency with absent categories.

3. Zero Directional Concentration. If the mean direction vector in Eq. (1) cancels to zero, then $\mathcal{M}_k^{c,c'} = 0$ and the normalized direction in Eq. (2) is undefined. In this case, we set all three directional components $\mathcal{C}_{k,x}^{c,c'}$, $\mathcal{C}_{k,y}^{c,c'}$, and $\mathcal{C}_{k,z}^{c,c'}$ to zero.

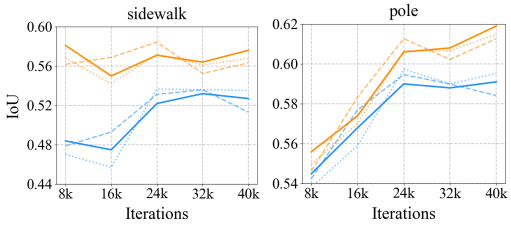

Figure 9: IoU evolution during training for geometrically structured classes on Synthia → Cityscapes, comparing our approach with the baseline (MIC). The individual curves represent results from three independent runs initialized with different seeds.

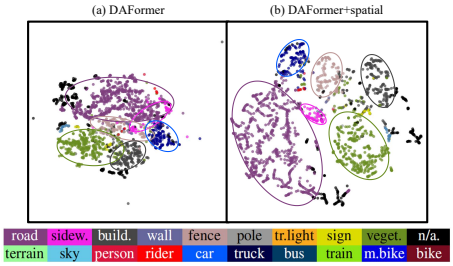

Figure 10: Feature visualization using t-SNE on the target domain. Compared to DAFormer (a), our method (b) significantly enhances inter-class separability, yielding clearer decision boundaries for semantic segmentation.

C Rationale for Global Geometric Aggregation

In Sec. 3.1, we aggregate class-wise statistics into a compact geometric descriptor (Eq. (3)). To assess this design, we compare class-aggregated and class-separated representations for directional concentration, dominant direction, and their combination. The aggregated variants are M8, M9, and M3, respectively; their class-separated counterparts are M14, M15, and M16. All comparisons use DAFormer on GTA→Cityscapes without pseudo-label weight adjustment. With $C = 19$ supporting classes, separating the combined cues produces 76 channels, whereas aggregation produces four.

geometry	Aggregated	Class-separated
	M8	M14
conc.	68.7 ± 0.6	67.6 ± 0.5
	M9	M15
dir.	69.0 ± 0.4	68.1 ± 0.5
	M3	M16
conc.+dir.	69.2 ± 0.3	68.4 ± 0.4

Table 7: Class-wise aggregation ablation. Results are mean ± std over three runs.

As shown in Table 7, aggregation outperforms class separation by 1.1, 0.9, and 0.8 for concentration, direction, and their combination, respectively. Class-separated targets preserve supporting-class identity but contain zero-valued channels when classes are absent, leaving fewer informative observations per channel and potentially increasing sensitivity to cross-domain differences in class frequency.

For each target instance, the final four-channel descriptor averages directional concentration and directional components across supporting classes. This reduces class-specific sparsity and provides a more consistent auxiliary signal across images. These results suggest that compact aggregation provides more effective auxiliary supervision in the evaluated setting.

D Cross-Domain Geometric Scale Consistency

The proposed geometric representation (Eqs. (1) to (3)) operates in a joint space of pixel coordinates (x, y) and normalized depth ($z \in [0, 255]$). To ensure numerical stability across the source and target domains, we employ fixed-size crops to strictly bound spatial coordinate ranges under two settings:

1. Low-Resolution Setting: Random 512×512 crops restrict spatial coordinates to $x, y \in [0, 512]$. This yields a comparable range with depth, maintaining a maximum scale ratio of approximately 2:1.

2. High-Resolution Setting: Random 1024×1024 crops expand the spatial bounds, maintaining a fixed scale ratio of approximately 4:1 relative to the depth.

By constraining spatial dimensions via cropping and normalizing depth, the input magnitudes for the geometric metric remain comparable. This helps prevent the constructed directional features from being dominated by any single dimension (x, y, or z). Furthermore, while the joint space inherently exhibits a specific scale ratio, we maintain a consistent static anisotropy across both domains rather than artificially forcing isotropy. Applying the same coordinate scaling convention in both domains avoids introducing an additional scale mismatch during geometric cue extraction. Empirical results confirm that our method remains robust under these variations, with negligible impact on the final segmentation performance.

E Source-domain Pretraining of Auxiliary Networks

For each benchmark, we pretrain a separate pair of auxiliary semantic segmentation networks using only the corresponding labeled source domain (GTA or Synthia). One network receives the one-channel concentration map and the other the three-channel direction map; both maps are constructed from source ground-truth labels and offline estimated depth. Each network predicts source semantic labels using cross-entropy loss. Both use an ImageNet-pretrained SegFormer encoder and a DAFormer decoder, without applying UDA strategies during pretraining. We train for 20k iterations with 512×512 crops and batch size 2, using the same optimizer, learning rate, and weight decay as DAFormer. The networks are then frozen during adaptation, where their forward predictions are used to adjust pseudo-label weights.

F Analysis of Learning Dynamics for Structured Classes

To further analyze the effect of the proposed geometric auxiliary tasks, we examine the evolution of validation IoU for spatially structured categories under inter-instance geometric constraints. Fig. 9 compares the learning dynamics of the baseline and our method on *sidewalk* and *pole*. As shown by the blue curves, the baseline underperforms our approach during most of the training process.

Consistent with the qualitative results in the main paper (Fig. 3), the baseline exhibits confusion between *sidewalk* and neighboring *road* regions, suggesting insufficient use of inter-instance spatial layout relations. For *pole*, the baseline often produces fragmented and spatially discontinuous predictions.

In contrast, our method (orange curves) maintains a clear and stable performance advantage. By explicitly modeling dominant inter-instance directions, the proposed approach effectively alleviates these geometric ambiguities. In particular, stable inter-class directional relations between *sidewalk* and *road* facilitate their separation based on relative spatial configuration rather than appearance similarity. For *pole*, inter-instance direction constraints provide contextual cues from surrounding regions, which help reduce fragmentation. These results suggest that the proposed inter-instance geometric cues act as structural regularizers, leading to more coherent predictions.

G Computational Efficiency Analysis

Method	Training		Inference	
	throughput	memory	throughput	memory
DAFormer	1.92 batch/s	9.0 GB	20.2 img/s	0.9 GB
+ spatial	1.51 batch/s	10.3 GB	19.1 img/s	0.9 GB
HRDA	0.84 batch/s	14.4 GB	1.6 img/s	8.8 GB
+ spatial	0.63 batch/s	16.6 GB	1.5 img/s	8.8 GB
MIC	0.68 batch/s	14.4 GB	1.6 img/s	8.8 GB
+ spatial	0.52 batch/s	16.6 GB	1.5 img/s	8.8 GB

Table 8: Training and inference throughput and GPU memory on GTA→Cityscapes using an RTX PRO 4500 GPU.

Table 8 reports the training and inference overhead. During training, adding our spatial module increases GPU memory by 1.3 GB for DAFormer and 2.2 GB for HRDA and MIC. Throughput decreases by approximately 21–25% across the three frameworks, reflecting the additional auxiliary models and multi-task computations. The two auxiliary segmentation models are used to adjust pseudo-label weights during adaptation and are not invoked at test time. During inference, reported GPU memory remains unchanged, while throughput decreases by approximately 5–6%. These results show that the method improves adaptation performance with moderate training overhead and a smaller inference cost.

H Feature Visualization with t-SNE

Fig. 10 visualizes the feature distributions in the target domain using t-SNE. While the baseline DAFormer exhibits noticeable class overlap, our method produces distinct and well-separated clusters. This indicates enhanced intra-class compactness and inter-class separability, suggesting that the proposed geometric priors effectively regularize the feature space for more discriminative representations.

I Component-wise Analysis of Dominant Direction

Method	Auxiliary type	Auxiliary detail	mIoU
M1	-	-	68.3 ± 0.5
M9	structure	dir. x, y, z	69.0 ± 0.4
M17	structure	dir. x only	68.5 ± 0.7
M18	structure	dir. y only	68.6 ± 0.5
M19	structure	dir. z only	68.8 ± 0.4

Table 9: We analyze the impact of using only the x, y, or z component of the dominant direction as the auxiliary task. All variants are trained without pseudo-label re-weighting to isolate the contribution of the auxiliary supervision.

To further investigate the effectiveness of the dominant direction cue, we decompose the inter-instance dominant direction vector into its three coordinate components (x, y, and z) and evaluate each component independently as an auxiliary supervision signal. Following the ablation protocol in the main paper, all variants are trained under a controlled setting where pseudo-label re-weighting is disabled. The results are summarized in Table 9.

As shown in the table, employing individual components of the inter-instance dominant direction yields comparable performance across M17–M19, with each providing a modest gain over the baseline:

- **Image-plane components (x and y).** M17 and M18 achieve 68.5 and 68.6 mIoU, respectively. These results suggest that inter-instance directional relations projected onto the image plane provide useful structural cues for semantic segmentation.
- **Depth-aligned component (z).** M19 achieves 68.8 mIoU, slightly higher than the image-plane components. This gain reflects the relative depth information captured by the z component of inter-instance directions.

The gap among the single-component variants is at most 0.3 mIoU, indicating comparable effects across the three dimensions. Using all three components (M9) achieves the highest mIoU in this comparison, suggesting that their combination provides complementary geometric information for unsupervised domain adaptation.

J Additional Qualitative Comparisons

Due to space limitations in the main manuscript, we provide additional qualitative comparisons in this section to further illustrate the performance of our approach. We visualize the segmentation predictions when our method is integrated into three state-of-the-art frameworks: DAFormer, HRDA, and MIC. The results cover both GTA→Cityscapes and Synthia→Cityscapes benchmarks. Specifically, the qualitative results for the DAFormer framework are presented in Fig. 11. Corresponding comparisons for the HRDA and MIC frameworks are illustrated in Fig. 12 and Fig. 13, respectively.

K Broader Societal Impacts

Our proposed geometry-aware Unsupervised Domain Adaptation method improves the robustness of semantic segmentation, offering significant positive impacts for autonomous systems like self-driving cars. However, we acknowledge potential negative impacts, including the intentional misuse of this technology for unauthorized surveillance or military drones, as well as the unintentional risk of physical harm if the model produces incorrect predictions in safety-critical scenarios. To mitigate these risks, we advocate for strict ethical guidelines governing computer vision deployment and emphasize the necessity of rigorous safety validation and uncertainty estimation before integrating such models into real-world applications.

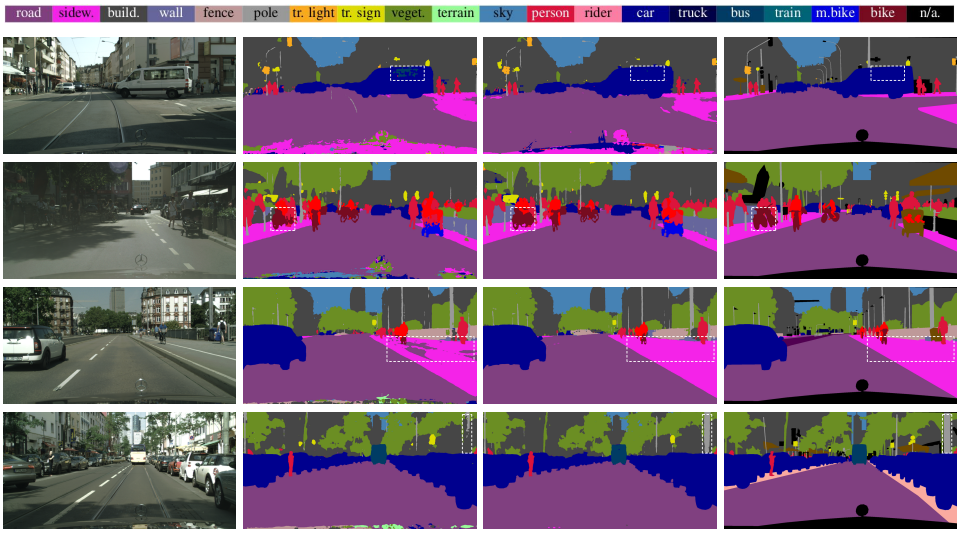

Figure 11: Qualitative results for Synthia→Cityscapes (top two rows) and GTA→Cityscapes (bottom two rows) when applied to the DAFormer framework. These results demonstrate the improvements of our approach compared to DAFormer. Notably, our method shows enhancements in the prediction of *car*, *bicycle*, *sidewalk*, *pole*. These enhancements are visually emphasized in the white boxes.



Figure 12: Qualitative results for Synthia→Cityscapes (top two rows) and GTA→Cityscapes (bottom two rows) when applied to the HRDA framework. These results demonstrate the improvements of our approach compared to HRDA. Notably, our method shows enhancements in the prediction of *sidewalk*, *pole*, *bicycle*, *fence*, *bus*. These enhancements are visually emphasized in the white boxes.

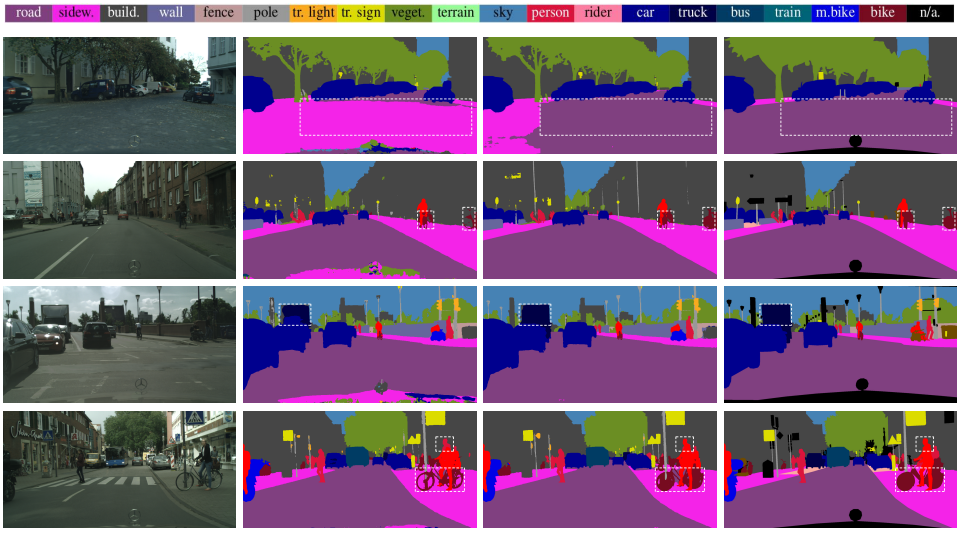

Figure 13: Qualitative results for Synthia→Cityscapes (top two rows) and GTA→Cityscapes (bottom two rows) when applied to the MIC framework. These results demonstrate the improvements of our approach compared to MIC. Notably, our method shows enhancements in the prediction of *road*, *bicycle*, *truck*, *rider*. These enhancements are visually emphasized in the white boxes.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction explicitly outline our core contributions, specifically detailing the proposed geometric representations, correlation loss, and pseudo-label weight refinement.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly discuss our limitations in the Conclusion section, specifically addressing the method's reliance on cross-domain consistent depth cues and its potential degradation in domains lacking such 3D information, such as medical imaging.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: This paper focuses on empirical contributions and does not include formal theoretical results or mathematical proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide comprehensive implementation details in the Experiments and Analysis sections. Furthermore, the source code and model checkpoints will be made publicly available to ensure full reproducibility.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will provide open access to our source code and model checkpoints, accompanied by detailed instructions. All experiments utilize standard, publicly available benchmark datasets.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We explicitly specify all necessary experimental configurations, including hyperparameters, optimizers, and dataset, within the Experiments and Analysis sections.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: To ensure statistical robustness, our main results report the mean and standard deviation of the mIoU evaluated over three independent runs with varying random seeds.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We detail the hardware specifications (e.g., GPU models) and provide comprehensive training logs that document the exact environment versions and computational overhead.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have carefully reviewed the NeurIPS Code of Ethics and confirm that all aspects of this research strictly adhere to its principles.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We explicitly discuss both the positive societal impacts on advancing autonomous systems and the potential negative risks associated with deploying such models in safety-critical applications.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: Not applicable, as our work does not introduce high-risk generative models or newly scraped datasets that would necessitate specific release safeguards.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All existing assets, including datasets and codebases, are properly cited, and their specific licenses (e.g., Apache 2.0) and terms of use are strictly respected.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Our newly released assets, including source code and pre-trained checkpoints, are accompanied by comprehensive documentation detailing installation, training protocols, and licensing.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Not applicable, as this research does not involve crowdsourcing or any experiments with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: Not applicable, as this research does not involve human subjects and therefore does not require Institutional Review Board (IRB) approval.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: Not applicable, as Large Language Models (LLMs) were not involved in any aspect of our core methodology or scientific contributions.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.