SCENE ADAPTATION AS RADIAL TRANSPORT IN HYPERBOLIC SPACE

Anonymous authors
Paper under double-blind review

ABSTRACT

Top-leading solutions for scene adaptation typically build on pseudo-label self-training with cross-domain mixing. Though demonstrating steady gains in mean IoU, they remain less reliable on small and rare categories: we observe that self-training improves large categories such as *truck* and *sky* while degrading *bicycle* and *rider*. In this paper, we reveal the underlying nature of this phenomenon: these approaches regulate how far features move but overlook their direction, and thus fall short in balancing domain alignment against label preservation. To this end, we introduce `RadDA`, which formulates scene adaptation as radial transport in hyperbolic space. Drawing inspiration from the observation that, in a low-dimensional Poincaré ball, the class of a pixel is encoded in its angle, rare categories occupy the narrowest sectors, and the domain shift is only partially radial, `RadDA` measures this trade-off by the angular action ε of the transport. Specifically, `RadDA`: i) couples source and target pixels via class-aware optimal transport, ii) learns a Riemannian flow under an explicit penalty on ε, and iii) gates out paths that turn beyond a pixel's angular margin. Since ε bounds both label corruption and the $O(\varepsilon/K)$ integration error, one integration step suffices and inference is unchanged. Extensive experiments with a frozen DINOv2 encoder demonstrate that `RadDA` improves tail-class mIoU on GTA→Cityscapes by 8.0 over the source-only head and 7.0 over self-training at higher mean IoU.

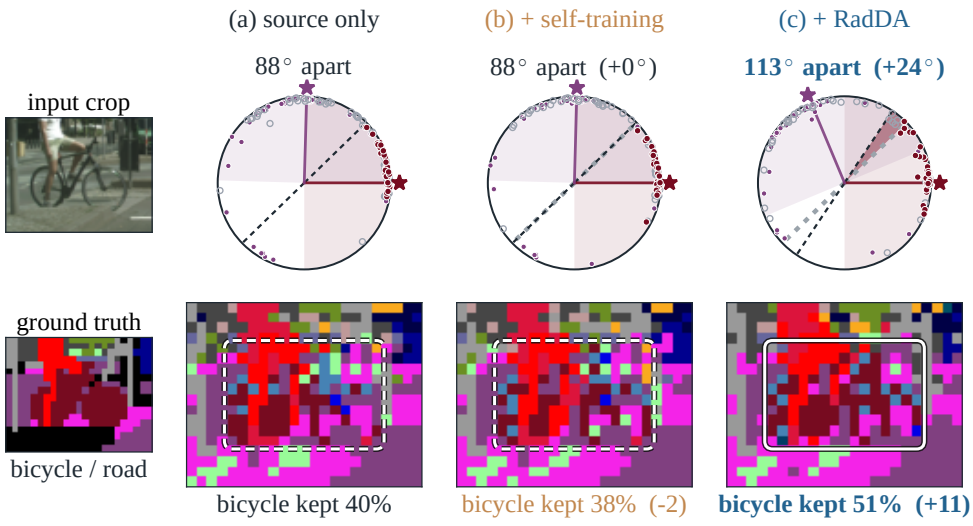

Figure 1: Three hyperbolic heads on the *same* frozen DINOv2-L features, applied to one Cityscapes crop in which a *bicycle* stands against *road*. **Top:** Poincaré disk of the two classes' prototypes ($\star$), each pixel at its own angle; dashed: decision boundary (grey: its position in (a)); filled / hollow: bicycle pixels kept / lost. **Bottom:** segmentation (bicycle boxed). **(a)** Source only: 88° apart, 40% kept. **(b)** Pseudo-label self-training leaves the geometry untouched: 88°, boundary moved by 0.0°, 38% kept, as the confident pseudo-labels here are *road* (on this head it does not raise mean IoU either, App. E). **(c)** `RadDA`, moving pixels along the radius and charging every degree they turn, widens the gap to 113° and swings the boundary 12° into the space of *road* (shaded): 51% kept.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Unsupervised scene adaptation transfers a semantic segmentation model from labeled source scenes to unlabeled target scenes, and top-leading solutions have advanced it steadily on the standard synthetic-to-real benchmark: self-training pipelines with cross-domain mixing raised GTA→Cityscapes mean IoU from 68.3 (DAFormer) to 75.9 (MIC) (Hoyer et al., 2022a;b; 2023), and adapting a vision foundation model lifts it to 79.8 (VFM-UDA++) (Englert & Dubbelman, 2025). The gains are not spread evenly. In our re-evaluation of VFM-UDA++, *road* and *car* reach 97.1 and 95.6 IoU while *pole* and *rider* stay at 66.7 and 68.2; in a controlled comparison on a frozen encoder, pseudo-label self-training raises mean IoU by 3.5 points, with gains of 11.7 on *truck* and 8.2 on *sky*, while *bicycle* IoU falls from 12.8 to 6.1. The small and rare categories are where an adapted segmenter remains least reliable, and they are the ones a driving system can least afford to miss (Richter et al., 2016; Sakaridis et al., 2021).

In this paper, we reveal the underlying nature of this phenomenon. Moving a pixel feature toward the target domain reduces the domain gap, yet the same move can carry the pixel across a class boundary, so every adaptation method trades domain alignment against label preservation. Existing methods settle this trade-off without controlling it, because they regulate how far features move and overlook the direction of the movement. **One-step** methods apply a single global correction, adversarially or by restyling the source (Tsai et al., 2018; Vu et al., 2019; Hoffman et al., 2018), and one displacement cannot bend around the boundary between a frequent category and a small one. **Multi-step** methods divide the move into many small corrections, through rounds of self-training, intermediate domains or generative transports (Zou et al., 2018; Dai et al., 2020; Kumar et al., 2020; Peng et al., 2023); each step appears safe and none is steered, so nothing prevents the steps from drifting a small-category pixel toward a frequent one. Methods designed for small categories re-sample them, re-threshold their pseudo-labels or re-weight their errors (Hoyer et al., 2022a; Zou et al., 2018; Xie et al., 2023), which changes how much training signal a category receives and leaves the direction of the move equally free. The cost of an unbalanced trade-off falls on the rare categories because they have the least room: a drift that keeps a road pixel on the road carries a bicycle pixel into it.

Fig. 1 shows this trade-off in one frame. A cyclist stands against the road, and a head trained on GTA labels recovers 40% of the cyclist's pixels. Pseudo-label self-training, the default remedy, supervises more target pixels but leaves the decision boundary between the two classes exactly where it was, and the cyclist keeps 38%, because the confident pseudo-labels in this crop belong to *road*. Over the validation set the same trade appears in the aggregate numbers above: self-training aligns the frequent categories and gives up the rare ones. A transport that controls its direction can instead spend angle where the bicycle needs room: `RadDA` widens the gap between the two prototypes from $88°$ to $113°$, and the cyclist keeps 51%.

To act on this trade-off, we rethink scene adaptation in two steps. ❶ We first locate the label relative to the shift. We embed pixel features of a frozen encoder in a low-dimensional Poincaré ball and split each feature into a radius and an angle (§3.1). A classifier that reads only the angle stays within one point of a Euclidean head (-0.3 and $+0.9$ mIoU on the two benchmarks), so class identity is angular, while the domain shift is only partly radial (51% and 60% of the class-mean displacement). The trade-off thus becomes geometric: radial motion reduces the domain gap at no cost to the label, the remaining gap requires angular motion, and angular motion is the only motion that can carry a pixel out of its class sector. Small categories own the narrowest sectors, precisely because they appear inside the context of frequent ones. ❷ We then control how a transport spends that angle. Diffusion and flow matching (Ho et al., 2020; Lipman et al., 2023) are the natural tools here, because they are the only family in which the angular part of a path can be written down, bounded and penalized. We call this quantity the *angular action* ε of a transport; it can be measured for any adaptation method, including one-step methods that never define a path, and penalizing it is how a transport can be steered along the trade-off.

We instantiate this as `RadDA`. A light head with learned curvature lifts both domains into the ball, where a conformal decomposition splits any tangent vector into a free radial part and a charged angular part in $O(d)$. `RadDA` then i) couples source and target pixels by class-aware optimal transport, so that a coupled pair is nearly radial before any penalty applies, ii) trains a class-conditioned flow by Riemannian flow matching under an explicit penalty on ε, and iii) gates out paths that turn past a pixel's angular margin, while a Busemann prototype classifier reads labels from the angle alone. The

single scalar ε yields three guarantees (§3): an angular margin for small categories that degrades only logarithmically with the number of classes, label validity along the whole path, and an integration error of $O(\varepsilon/K)$ that makes one step sufficient. `RadDA` thus has the continuity of multi-step transfer at the cost of one step, and it exists only at training time.

Our work exhibits several compelling facets. **(i)** It makes the trade-off measurable: on DarkZurich every flow or diffusion transport without the angular penalty lands at $\varepsilon \geq 1.09$ and gains at most 0.3 mIoU, raising the penalty moves `RadDA` along the trade-off to an optimum in between, and there `RadDA` spends about half the angular action of an unsteered Euclidean flow on the same coupling (0.71 against 1.22–1.40) and ends above the source-only head where that flow ends 1.1–2.2 below it. **(ii)** It is cheap: one integration step is as good as fifty, a step costs the same as one step of a diffusion transport, against which `RadDA` is 1.4 mIoU better on DarkZurich, and test-time inference is the bare head. **(iii)** It recovers the classes the mean hides: with a frozen DINOv2 encoder on GTA→Cityscapes, `RadDA` raises tail mIoU by 8.0 over the source-only head and by 7.0 over self-training at 1.2 points higher mean IoU, taking *bicycle* from 12.8 to 33.4 where self-training takes it to 6.1. **(iv)** The same experiments delimit it: on DarkZurich the tail does not improve, the pattern fails on the weaker ViT-S encoder, and on an already self-trained host plain self-training wins (§4.4).

2 Related Work

Scene adaptation. Early methods aligned the domains in a single correction, adversarially in feature or output space (Ganin et al., 2016; Tsai et al., 2018; Vu et al., 2019) or by restyling source images (Hoffman et al., 2018; Yang & Soatto, 2020). Self-training then became dominant (Zou et al., 2018; Araslanov & Roth, 2021; Tranheden et al., 2021; Zhang et al., 2021) and underpins DAFormer, HRDA and MIC (Hoyer et al., 2022a;b; 2023), which pair a SegFormer backbone (Xie et al., 2021) with cross-domain mixing, multi-resolution crops and masked consistency; contrastive variants add pixel- or patch-level objectives (Xie et al., 2023; Chen et al., 2023a) and curriculum methods pass through intermediate domains (Sakaridis et al., 2019; Dai et al., 2020; Kumar et al., 2020). All of them regulate how far a representation moves and leave its direction free, so none controls the trade-off between domain alignment and label preservation on which the rare classes depend.

Generative transports, flow matching and optimal transport. Diffusion models synthesize target-like training images (Peng et al., 2023; Benigmim et al., 2023; Niemeijer et al., 2024; Jia et al., 2024; Wu et al., 2023), project corrupted test inputs back onto the source distribution (Gao et al., 2023), and through flow matching learn image-to-image translations (Le Bellier & Audebert, 2026; VERIFY-AUTHORS, 2026a); diffusion features are strong dense descriptors (Baranchuk et al., 2022; Xu et al., 2023; Tian et al., 2024) and diffusion decoders have been used for dense prediction (Amit et al., 2021; Ji et al., 2023; Chen et al., 2023b). All of this operates in image space and pays the K-step sampling cost at inference. Underneath, flow matching learns a velocity field from paired endpoints (Lipman et al., 2023; Liu et al., 2023), minibatch optimal-transport couplings straighten the paths (Pooladian et al., 2023; Tong et al., 2024), Riemannian flow matching extends this to manifolds (Chen & Lipman, 2024), and optimal transport is long established in domain adaptation (Courty et al., 2017; Guo et al., 2023). We place a continuous transport in feature space instead, for a property unrelated to generation: it is the only family in which the angular part of the motion can be written down, bounded and penalized. Our coupling is class-aware rather than purely geometric, and the field carries an explicit penalty on its angular component. Feature-space flow adapters for vision-language alignment (VERIFY-AUTHORS, 2025a; 2026c) make the same case for continuous over one-step adaptation, but for few-shot classification and with no step-count guarantee.

Hyperbolic representations. Hyperbolic embeddings encode hierarchy with low distortion (Nickel & Kiela, 2017; Ganea et al., 2018; Khrulkov et al., 2020) and have been applied to segmentation (Ghadimi Atigh et al., 2022), metric learning (Ermolov et al., 2022) and image-text alignment (Desai et al., 2023). HALO (Franco et al., 2024) turns the radius, which tracks data scarcity under shift, into an active-learning score; RAFT (VERIFY-AUTHORS, 2025b) adapts hyperbolic feature augmentation to segmentation, and recent work decouples radius and angle for open-vocabulary segmentation (VERIFY-AUTHORS, 2026b). We use the same split for a different purpose: the angle carries the label, so it also defines a budget, and what we design is the transport that spends it.

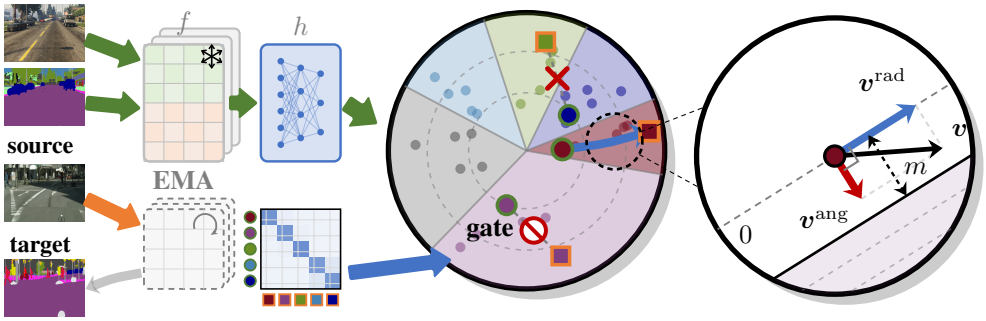

Figure 2: RadDA lifts both domains into one Poincaré ball and moves source pixels along the radius while charging every degree they turn. *(1) Frozen encoder and teacher.* f maps a labeled source and an unlabeled target frame to patch features, an EMA copy supplies pseudo-labels, and a light head h lifts both into the ball. *(2) Class-aware coupling.* A Sinkhorn plan biased by those pseudo-labels pairs each source pixel with target pixels of the same class (Eq. 5), so a coupled pair is nearly radial before any penalty applies. *(3) Penalized transport.* The classifier reads the label from the angle alone, so the ball is partitioned into class sectors, wide for *road* and narrow for *bicycle*; the field carries each source pixel (circles) to its coupled target pixels (squares), and the callout splits one move into a free $\boldsymbol{v}^{\mathrm{rad}}$ and an $\boldsymbol{v}^{\mathrm{ang}}$ charged against ε (Eq. 2). *(4) Sector gate.* A path that turns past its own margin is dropped from the loss (Eq. 7). Nothing here reads a target label or runs at test time.

3 Angular-Budgeted Transport in the Poincaré Ball

This section first measures where the label and the domain shift live in a hyperbolic embedding, which turns the trade-off of §1 into two measurable quantities (§3.1), and then builds the transport that follows from the two measurements (§3.2).

3.1 Class is Angular, Shift is Only Partly Radial

A frozen DINOv2-L/14 encoder (Oquab et al., 2024), the starting point of recent VFM-based scene adaptation (Wei et al., 2024; Englert & Dubbelman, 2025), gives a feature per pixel; a light head maps it into a Poincaré ball ($d=32$, learned curvature) with the angular classifier of §3 on top, trained on source labels only (App. D).

Class is angular. Replacing a Euclidean head by a hyperbolic head whose classifier depends on x *only through its angle* changes source-trained target mIoU by -0.3 points on Cityscapes→DarkZurich and $+0.9$ on GTA→Cityscapes: the label is readable from the direction, so a transport that alters the angle risks it while one that alters only the radius does not. The tail is where the two heads differ: the angular head is 2.8 tail points above the Euclidean head on GTA→Cityscapes and 4.1 below it on DarkZurich, where the narrow sectors of the rare classes cost them before any transport is applied.

Shift is not purely radial. Splitting each class's source-to-target mean displacement into radial and angular parts gives a class-averaged radial fraction of 0.51 on Cityscapes→DarkZurich and 0.60 on GTA→Cityscapes (three seeds each): radial-dominant on the synthetic-to-real gap, balanced on the day-to-night gap, purely radial on neither. Together the two measurements make the trade-off of §1 geometric: radial motion reduces the gap at no cost to the label, *some* angular motion is needed to close the rest, and every unit of it is spent against the label margin, so a transport has to budget it.

Measuring what a transport does. For any method M mapping a source pixel embedding x to an adapted embedding $M(x)$, we define its *angular action*

$$\varepsilon(M) \;=\; \mathbb{E}_{x\sim\mathcal{D}_S}\Big[\int_0^1 \big\|\boldsymbol{v}_t^{\mathrm{ang}}(x)\big\|_g \, dt\Big], \tag{1}$$

the expected angular path length in the hyperbolic metric; for one-step methods the path is the geodesic from x to $M(x)$. We also record the *class-crossing rate* $\rho(M)$, the fraction of transported pixels whose predicted class differs from their label somewhere along the path. Fig. 3(a) places every transport we can run under identical conditions by its ε (Table 2 gives the same runs with ρ and tail IoU), and §4.3 reads the result: what separates the transports that help from those that do not is a constraint on the direction of the steps, and their number plays no role.

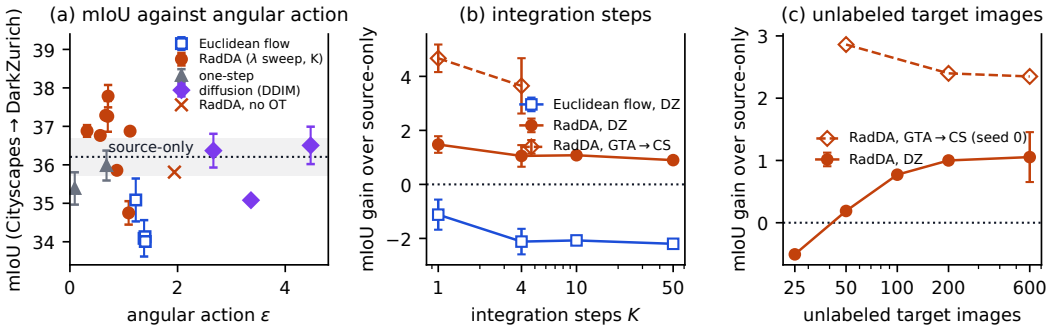

Figure 3: (a) mIoU against measured angular action ε for every transport we run under identical conditions in the frozen-encoder protocol on Cityscapes→DarkZurich, including the λ_{ang} sweep of `RadDA` (Table 2 gives ε, ρ and tail IoU; error bars are the seed spread where three seeds exist); the dotted line and band are the source-only head. (b) mIoU gain over the source-only head against the number of integration steps K for the unsteered Euclidean flow and for `RadDA` (DarkZurich, filled; GTA→Cityscapes with gate and EMA, open diamonds, three seeds): more steps do not help the unsteered flow, and one step is best for `RadDA`. (c) Gain against the number of unlabeled target images ($K = 4$; seed 0, except the three-seed DarkZurich point at 600): on DarkZurich it grows up to about 200 images, and on GTA→Cityscapes 50 images give as much as 600.

3.2 Radial Transport with an Angular Budget

`RadDA` inserts a transport between the encoder and the classifier of a scene-adaptation pipeline, and it exists only at training time (Fig. 2). Its design follows the two measurements of §3.1: the label lives in the angle, and the shift is mostly, though not entirely, radial. A single scalar governs the whole construction, *i.e.*, the angular action ε of the learned flow, which sets how much label margin the transport spends to reduce the domain gap. Below we first decompose a move into the part that is free and the part that is charged (Eq. 2), then show what the charge buys in label validity (Eqs. 3–4), and finally detail the three components that spend it: a class-aware coupling (Eq. 5), an angle-penalized flow (Eq. 6) and a sector gate (Eq. 7). Formal statements and proofs are collected in App. C.

Radial–Angular Decomposition of a Move. A light head h_ϕ ($D \to 256 \to d$, $d = 32$) followed by the exponential map at the origin lifts a pixel feature $\boldsymbol{z}_p = f_\theta(I)_p$ of either domain into the Poincaré ball, $\boldsymbol{x}_p = \exp_0^c(h_\phi(\boldsymbol{z}_p)) \in \mathbb{B}_c^d$. The ball carries the metric $g_{\boldsymbol{x}} = \lambda_{\boldsymbol{x}}^2 \, g^E$ with $\lambda_{\boldsymbol{x}} = 2/(1 - c\|\boldsymbol{x}\|^2)$ (Ganea et al., 2018), its curvature c is learned, radii are clipped clear of the numerically unstable boundary (Mishne et al., 2023), and the hyperbolic radius of $\boldsymbol{x}$ is $r(\boldsymbol{x}) = \frac{2}{\sqrt{c}} \operatorname{artanh}(\sqrt{c}\|\boldsymbol{x}\|)$. Since the metric is conformal, *i.e.*, Euclidean angles at a point are hyperbolic angles, every tangent vector $\boldsymbol{v} \in T_{\boldsymbol{x}}\mathbb{B}_c^d$ splits along the outward direction $\boldsymbol{u} = \boldsymbol{x}/\|\boldsymbol{x}\|$ into two $g_{\boldsymbol{x}}$-orthogonal parts, and integrating the second of them along a path gives the quantity this paper is written in:

$$\boldsymbol{v} = \underbrace{\langle \boldsymbol{v}, \boldsymbol{u} \rangle \, \boldsymbol{u}}_{\boldsymbol{v}^{\text{rad}}:\ \text{free}} + \underbrace{\boldsymbol{v} - \langle \boldsymbol{v}, \boldsymbol{u} \rangle \, \boldsymbol{u}}_{\boldsymbol{v}^{\text{ang}}:\ \text{charged}}, \qquad \varepsilon(\boldsymbol{x}_0) = \int_0^1 \left\| \boldsymbol{v}_t^{\text{ang}} \right\|_g \mathrm{d}t, \tag{2}$$

where $\boldsymbol{v}_t^{\text{ang}}$ is the angular velocity at time t along the path leaving $\boldsymbol{x}_0$, and the average of $\varepsilon(\boldsymbol{x}_0)$ over source pixels is the ε of Fig. 3. Being an inner product and a subtraction, the split costs $O(d)$ per pixel and calls neither the exponential nor the logarithmic map. Note that ε is a *hyperbolic length* rather than an angle, which is what makes the comparison in Eq. 3 well posed.

Class Sectors and the Angular Margin. The classifier is a Busemann prototype head (Ghadimi Atigh et al., 2021): class k owns an ideal prototype $\boldsymbol{\omega}_k \in \mathbb{S}^{d-1}$ on the boundary and $\boldsymbol{x}$ is scored by $-B_{\boldsymbol{\omega}_k}(\boldsymbol{x})$, with $B_{\boldsymbol{\omega}}(\boldsymbol{x}) = \log(\|\boldsymbol{\omega} - \boldsymbol{x}\|^2/(1 - \|\boldsymbol{x}\|^2))$. Between two classes the radius enters both logits as a common scale, so the comparison depends on $\boldsymbol{x}$ only through its direction and each decision region is a *cone* from the origin. Every class thus owns an angular sector, whose room we measure not in degrees but in the same metric ε lives in:

$$m(\boldsymbol{x}_0) = \tfrac{1}{\sqrt{c}} \operatorname{arsinh}\big(\sinh(\sqrt{c}\, r(\boldsymbol{x}_0)) \, \sin(\alpha/2)\big), \tag{3}$$

where α is the angle subtended at the origin by the two prototypes flanking $\boldsymbol{x}_0$'s sector, so that $m(\boldsymbol{x}_0)$ is the hyperbolic distance from $\boldsymbol{x}_0$ to the nearest boundary of its own sector. Intuitively, Eq. 3 says a

sector of *fixed* angular width is worth more room the further out a pixel sits, and that a narrow sector, which is what a rare class owns, is short of room at every radius; the two prototypes of Fig. 1 subtend $88°$ before adaptation and $113°$ after.

Because $\varepsilon(\boldsymbol{x}_0)$ and $m(\boldsymbol{x}_0)$ are both hyperbolic lengths they compare directly, and that comparison is the guarantee the method rests on. If $\varepsilon(\boldsymbol{x}_0) < m(\boldsymbol{x}_0)$ the path cannot reach a sector boundary, so the head returns the label of $\boldsymbol{x}_0$ at *every* point along it, the endpoint at which the classifier is trained included. Taking expectations bounds the class-crossing rate ρ, *i.e.*, the fraction of transported pixels whose training label has silently gone wrong:

$$\rho \;\le\; \Pr_{\boldsymbol{x}_0 \sim \mathcal{D}_S}\big[\varepsilon(\boldsymbol{x}_0) \ge m(\boldsymbol{x}_0)\big] \;\le\; \frac{\mathbb{E}[\varepsilon]}{\mathbb{E}_{\text{tail}}[m]}, \tag{4}$$

by Markov's inequality on the tail classes. Eq. 4 is what makes ε worth penalizing: it is the numerator of the only quantity that can corrupt the training signal without ever showing up in the loss.

Class-Aware Entropic Coupling. Flow matching needs pairs $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ of a source and a target embedding, and the pairing decides which sectors the path has to cross. Independent pairing produces crossing paths, and plain minibatch optimal transport on pixel features (Tong et al., 2024) is blind to class structure, pairing a *rider* with a *person* whose sectors are adjacent. We therefore solve an entropic transport problem (Cuturi, 2013) under a cost that is hyperbolic distance plus a class term,

$$C(\boldsymbol{x}_p^S, \boldsymbol{x}_q^T) = d_c(\boldsymbol{x}_p^S, \boldsymbol{x}_q^T) + \beta \underbrace{\mathbf{1}\big[y_p \neq \hat{y}_q\big] \cdot \mathbf{1}\big[\max_k \hat{p}_q(k) \ge \tau\big]}_{\text{active only where the pseudo-label can be trusted}}, \tag{5}$$

where y_p is the source label, $\hat{y}_q$ the target pseudo-label of the current classifier or of the host's teacher when a host is present, and $\tau = 0.9$ the confidence threshold. Wherever that pseudo-label can be trusted the plan is restricted to a single sector, which makes the geodesic between a coupled pair *nearly radial before any penalty applies*. Pixels are subsampled per class, so the plan is class-balanced rather than $O(HW)$.

Angle-Penalized Riemannian Flow. For a coupled pair the reference path is the geodesic $\gamma_t(\boldsymbol{x}_0, \boldsymbol{x}_1)$ of the ball, and a velocity network $\boldsymbol{v}_\psi(\boldsymbol{x}, t, y)$ conditioned on the (pseudo-)class is trained by Riemannian flow matching (Chen & Lipman, 2024) against the conditional target $\dot{\gamma}_t = \log_{\gamma_t}(\boldsymbol{x}_1)/(1-t)$, which recovers the marginal field of the coupling in expectation (Lipman et al., 2023; Chen & Lipman, 2024). On top of it we charge the angular component,

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t,(\boldsymbol{x}_0,\boldsymbol{x}_1)} \big\| \boldsymbol{v}_\psi(\gamma_t, t, y) - \dot{\gamma}_t \big\|_{g_{\gamma_t}}^2, \qquad \mathcal{L}_{\text{ANG}} = \mathbb{E}_{t,(\boldsymbol{x}_0,\boldsymbol{x}_1)} \big\| \boldsymbol{v}_\psi^{\text{ang}}(\gamma_t, t, y) \big\|_{g_{\gamma_t}}, \tag{6}$$

so that $\mathcal{L}_{\text{ANG}}$ is a one-sample estimate of ε in Eq. 2 and λ_{ang} sets how strictly the path is held to the radius. The coupling having already removed most cross-sector pairs, this penalty suppresses residual turning rather than fighting the data; the two are *not* interchangeable, and without the coupling it leaves ε near 2 (§4.3). The transport runs source to target because only source pixels carry a label: carrying them into the region the target occupies is what lets the classifier be trained, with known labels, where it will be used.

Sector Gate. Eq. 4 bounds label corruption *on average*; applied pixelwise it removes it exactly where it happens. The term that trains the classifier on transported pixels is therefore evaluated only inside the sector,

$$\mathcal{L}_{\text{TR}} = \mathbb{E}_{(\boldsymbol{x}_0,y)\sim\mathcal{D}_S}\Big[\underbrace{\mathbf{1}\big[\varepsilon(\boldsymbol{x}_0) < m(\boldsymbol{x}_0)\big]}_{\text{sector gate}} \cdot \ell\big(\text{cls}(\boldsymbol{x}_1^{S\to T}),\, y\big)\Big], \tag{7}$$

with ℓ the per-pixel cross-entropy and $\boldsymbol{x}_1^{S\to T}$ the endpoint of the transport. In practice we test the net angle travelled against half the angle to the nearest other prototype, a quantity already computed during integration; this lower-bounds $\varepsilon(\boldsymbol{x}_0)$, so the gate relaxes rather than tightens Eq. 7. It retains 50% of transported pixels on GTA→Cityscapes and 77% on DarkZurich at $K = 4$, and the gain decays without it (§4.3).

Why One Step Suffices. Multi-step transports pay K network evaluations per pixel because each step must be small for the chain to stay safe; a path that barely turns removes that reason. A K-step geodesic Euler integrator has endpoint error $O(\varepsilon/K)$, with a constant depending on the Lipschitz constant of $\boldsymbol{v}_\psi^{\text{ang}}$ *alone*, so the step count is set by how much angle a transport spends and not by

Table 1: GTA→Cityscapes under the frozen-encoder protocol (DINOv2-L frozen, d=32 head, final iterate); every row is the same head on the same features and differs only in how it is adapted. **Geo.**: $\mathbb{E}$ Euclidean, $\mathbb{H}$ hyperbolic; K: transport steps (0 = no transport). **Best**, second best; $_{\uparrow}$ is the gain over the three-seed mean of the Euclidean source-only head. Subscripts are the population standard deviation over three seeds; **a cell without one is seed 0 alone**, which includes the full-resolution entries of our K=1 row. Backbone-training pipelines reach 68.3 (DAFormer), 75.9 (MIC) and 79.8 (VFM-UDA++, re-evaluated by us).

			patch-level		full resolution	
Method	Geo.	K	mIoU↑	tail↑	mIoU↑	tail↑
source-only head (Euclidean)	$\mathbb{E}$	0	$49.9_{\pm 0.4}$	$32.7_{\pm 0.9}$	$51.8_{\pm 0.3}$	$32.0_{\pm 0.6}$
source-only head (angular)	$\mathbb{H}$	0	$50.8_{\pm 0.6}$	$35.5_{\pm 1.9}$	52.7	34.7
self-training, Euclidean head (τ=0.9)	$\mathbb{E}$	0	$53.4_{\pm 0.1}$	$33.7_{\pm 0.4}$	53.9	31.6
prototype shift	$\mathbb{H}$	1	$50.1_{\pm 0.7}$	$33.1_{\pm 3.7}$	52.8	34.3
adversarial adapter	$\mathbb{H}$	1	$50.5_{\pm 0.6}$	$33.6_{\pm 1.6}$	52.5	33.3
Euclidean flow	$\mathbb{E}$	4	$49.5_{\pm 0.5}$	$34.3_{\pm 1.4}$	50.3	32.3
feature-space diffusion	$\mathbb{E}$	50	$50.8_{\pm 0.6}$	$31.5_{\pm 1.8}$	51.6	28.2
closed-form radial step ($\boldsymbol{v}^{\text{ang}} \equiv \mathbf{0}$)	$\mathbb{H}$	1	$50.6_{\pm 0.5}$	$34.8_{\pm 2.0}$	53.1	36.0
RadDA, no gate, no EMA	$\mathbb{H}$	4	50.1	30.4	53.1	32.4
RadDA, gate + EMA	$\mathbb{H}$	4	$53.6_{\pm 1.0}$	$37.3_{\pm 4.1}$	$55.8_{\pm 0.9}$	$38.4_{\pm 3.6}$
RadDA, gate + EMA	$\mathbb{H}$	1	**54.6**$_{\pm 0.5}$ $_{\uparrow 4.7}$	**40.7**$_{\pm 2.4}$ $_{\uparrow 8.0}$	**56.0** $_{\uparrow 4.2}$	**38.9** $_{\uparrow 6.9}$

how far it travels; at the value our flow reaches, $K=1$ already matches $K=10$. In the limiting case $\boldsymbol{v}_{\psi}^{\text{ang}} \equiv \mathbf{0}$ the flow never leaves the ray through $\boldsymbol{x}_0$, every K is exact, and the endpoint is closed-form, $\boldsymbol{x}_1 = \tanh(\frac{\sqrt{c}}{2} r_1)\boldsymbol{u}(\boldsymbol{x}_0)/\sqrt{c}$; implemented as a two-layer radial network it preserves the label exactly and recovers only part of the gain.

Overall Objective. RadDA trains jointly with the host pipeline,

$$\mathcal{L} = \underbrace{\mathcal{L}_{\text{SEG}}^{S} + \mathcal{L}_{\text{SEG}}^{T,\text{pseudo}}}_{\text{host self-training}} + \lambda_{\text{FM}}\mathcal{L}_{\text{FM}} + \underbrace{\lambda_{\text{ang}}\mathcal{L}_{\text{ANG}}}_{\text{the budget}} + \underbrace{\lambda_{\text{tr}}\mathcal{L}_{\text{TR}}}_{\text{gated, detached}}, \tag{8}$$

where the transport is detached in the last term so the flow cannot move pixels to wherever the classifier happens to be right already, and the host's teacher, mixing and consistency losses are untouched. One scalar therefore carries three roles across Eqs. 4, 6 and 7: the margin a path consumes, the labels it corrupts, and the integration error at a given step count. App. C assembles them into a target risk bound in which the $\mathcal{H}\Delta\mathcal{H}$-divergence of Ben-David et al. (2010) is replaced by quantities measurable on the training data; App. E gives pseudo-code and hyper-parameters.

4 Experiments

4.1 Setup

Protocols and benchmarks. Because the claims of §3 concern *how* features are transported, we hold everything else fixed: a frozen DINOv2-L/14 encoder (Oquab et al., 2024) and a light head trained on source labels, with or without a transport. This *frozen-encoder protocol*, where Rein (Wei et al., 2024) and VFM-UDA (Englert & Dubbelman, 2025) start, isolates the transport from self-training, augmentation and decoder effects. Benchmarks are GTA (Richter et al., 2016)→Cityscapes (Cordts et al., 2016) and Cityscapes→DarkZurich (Sakaridis et al., 2019); we report patch-level mIoU of the final iterate (600 source and 600 unlabeled target images unless stated) and full-resolution mIoU for the main GTA→Cityscapes rows, with *tail mIoU* the mean over *train*, *rider*, *motorcycle*, *bicycle* and *pole*; ± is the population standard deviation over three seeds. The second protocol freezes a *trained* host (DAFormer, MIC) and substitutes its decoder features and teacher pseudo-labels for the DINOv2 ones (§4.4). The controls are the alternatives of §3.1 (Table 1); architecture, hyper-parameters and pseudo-code are in App. E.

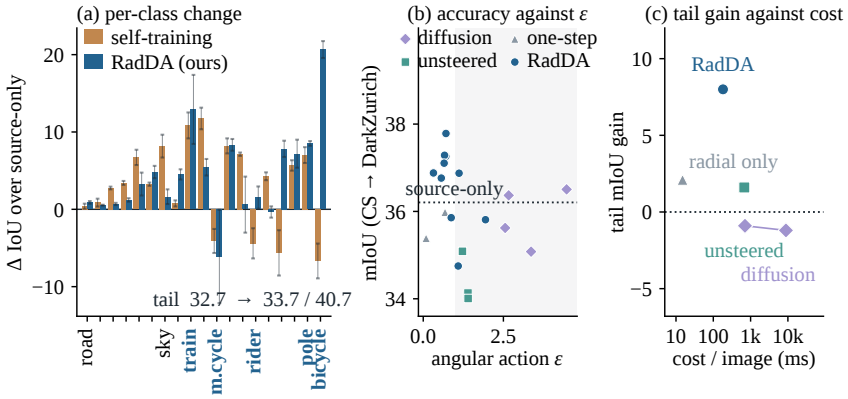

Figure 4: **(a)** Per-class IoU change over the Euclidean source-only head on GTA→Cityscapes (the runs of Table 1; tail classes in bold). Self-training gains on *truck* and *sky* and loses on *bicycle*, *rider*, *motorcycle* and *traffic sign*, leaving the tail at 33.7; `RadDA` takes it from 32.7 to 40.7 (three-seed tail means of Table 1; the bars are relative to seeds 1–2 of the source-only head) and also loses on *motorcycle*. **(b)** The DarkZurich transports of Table 2 placed by ε: the flow and diffusion transports without an angular penalty sit at $\varepsilon > 1$ (shaded) and gain at most 0.3 mIoU; the penalized flows with the class-aware coupling end no lower than their starting head. **(c)** GTA→Cityscapes tail gain against the cost of transporting one 2-megapixel image: $+8.0$ at 0.18 s for `RadDA`, -0.9 and -1.2 for diffusion at 4 (seed 0) and 50 steps.

4.2 Comparison Under a Fixed Encoder

Synthetic-to-real. In Table 1, `RadDA` lifts the frozen-encoder head from 49.9 ± 0.4 to 54.6 ± 0.5 mIoU and its tail from 32.7 ± 0.9 to 40.7 ± 2.4 ($K = 1$); at full resolution, from 51.8 and 32.0 (three seeds) to 56.0 and 38.9 (seed 0). Every other *transport* on the same head stays within one point of source-only, from -0.4 for the unsteered flow to $+0.9$ for the diffusion transport, which also has the lowest tail of these controls at 31.5. The one control that is not a transport, pseudo-label self-training with the same teacher and threshold, reaches 53.4 ± 0.1 mIoU but only 33.7 ± 0.4 tail. Fig. 4(a) is that difference per class: the two methods are 1.2 points apart on the average and 7.0 apart on exactly the classes this paper is about, self-training taking *bicycle* from 12.8 to 6.1 where `RadDA` takes it to 33.4; both lose on *motorcycle*, `RadDA` more (-6.1 against -4.1). The tail gain carries the most seed variance: at $K = 4$ it is $+4.6$ against a spread of ± 4.1, so we rest the claim on $K = 1$ ($+8.0$ against ± 2.4). Two ingredients keep it alive: without the sector gate and the EMA teacher `RadDA` peaks at 53.0 and decays to 50.1; with both, the final iterate is within 0.2 mIoU of the best one at $K = 1$ and within 0.6 at $K = 4$.

Day-to-night. On Cityscapes→DarkZurich (Table 5) `RadDA` improves the Euclidean source-only head by $+1.6$ mIoU at $K = 1$ over three seeds and self-training by $+1.0$, while every other transport lands between 2.2 below the source-only head and 0.3 above it (Tables 5 and 2). It does not improve the tail here: at 23.9 ± 0.4 it recovers 2.6 of the 4.1 tail points the angular head gives up, but stays 1.5 below the Euclidean source-only head and 1.9 below self-training (App. H).

4.3 The Angular Action Across Transports

This is the central experiment: encoder, coupling and classifier are fixed and only the transport varies (Fig. 3; Table 2 has the numbers). Three regularities emerge on DarkZurich. *First*, an unsteered flow is worse than no flow and more steps do not save it (Fig. 3b): Euclidean flow matching on the same coupling ends 1.1 to 2.2 mIoU below the Euclidean source-only head at $K = 1, 4, 50$ while its angular action *grows* with K ($1.22 \to 1.40$), so the drift comes from the field itself. *Second*, the penalty has an interior optimum: raising λ_{ang} through $0, 1, 2, 4, 8$ at $K = 1$ lowers ε monotonically from 1.09 to 0.32, while mIoU runs 34.8, 35.9, 37.8, 36.8, 36.9 ($\lambda_{\text{ang}} = 1$ and 4 are single seeds). This is the trade-off of §1 measured directly: with too little penalty the flow drifts across sectors, and with too much it cannot spend the angular part of the shift that §3.1 says is there. The optimum depends on the domain pair (on GTA→Cityscapes $\lambda_{\text{ang}} = 8$ does better, App. A); we keep $\lambda_{\text{ang}} = 2$ on both benchmarks. *Third*, the penalty separates the transports that help from those that do not (Fig. 3a): every flow or diffusion transport without it lands at $\varepsilon \geq 1.09$ and none gains more than 0.3 mIoU over

the head it started from, diffusion at one step and at fifty included; every flow with the penalty, the class-aware coupling and the Busemann head ends no lower than its starting head ($\lambda_{\text{ang}} = 1$ ties it), at ε between 0.30 and 1.12. The value of ε alone does not decide the outcome: $\lambda_{\text{ang}} = 0$ ($K = 1$) and 0.5 ($K = 4$) reach nearly the same action (1.09, 1.12) and end 2.1 mIoU apart, the adversarial adapter spends $\varepsilon = 0.68$ and gains 0.1, and the closed-form radial step ($\varepsilon = 0$) recovers little. Nor does the class-crossing rate follow ε ($\rho = 0.40$–0.46 across the sweep, App. A).

Why the gate is needed. Without gate and teacher the gain on GTA→Cityscapes peaks early and decays ($53.0 \to 50.1$): the flow becomes a faithful copy of the coupling, a transported pixel lands on its coupled target pixel, and $\mathcal{L}_{\text{TR}}$ turns into self-training with a second pseudo-labeller and its confirmation bias. With gate and teacher, 50% of transported pixels are kept on GTA→Cityscapes and the decay largely disappears; on DarkZurich the gate alone keeps 77% at $K = 4$ and changes nothing.

Coupling. The coupling is what directs the angle. Replacing the OT plan by independent pairing raises ε from 0.70 to 1.94 and drops mIoU below the Euclidean source-only head (35.8). Its class term matters only where pseudo-labels do: on DarkZurich it has no measurable effect (seed 0: 37.2 with $\beta_1 = 2$, 37.3 without) and with the EMA teacher it never changes the plan, so the two ablation rows coincide (Table 6); on GTA→Cityscapes gate and teacher are worth +3.5 mIoU and +6.9 tail (Table 1, $K = 4$).

Encoder, data and cost. The pattern holds on ViT-B (+8.9 tail over self-training) and ViT-L and fails on ViT-S (App. F, with the head geometry, curvature, dimension and target-set analyses). One step of any learned flow or diffusion transport costs about 0.18 s per 2-megapixel image and nothing at test time; at that cost `RadDA` leads one-step diffusion by 1.4 mIoU on DarkZurich, and fifty diffusion steps (8.9 s) add nothing.

4.4 Where the Gain Stops: an Already-Adapted Host

We freeze the released DAFormer GTA→Cityscapes model, already self-trained for 40k iterations (68.7 in our evaluation, 68.3 published), and substitute its decoder features and its own pseudo-labels for the DINOv2 ones.

The ordering reverses. On the host's features `RadDA` still improves on the source-only head, the tail more than the mean (64.2 ± 0.5 mIoU / 48.2 ± 1.7 tail against 62.2 ± 0.3 / 45.4 ± 1.0 at full resolution), but self-training on those same features reaches 67.7 ± 0.1 / 55.0 ± 0.1, 3.5 points above `RadDA`, where under the frozen-encoder protocol `RadDA` led it by 1.2 mIoU and 7.0 tail points; on the stronger MIC host the gain disappears on mean IoU (64.2 against 64.3) and becomes a 1.8-point loss on the tail. Such a host has already removed most of the angular gap of §3.1 and supplies far better pseudo-labels than our head can, so the simplest consumer of those labels wins. The contribution is therefore scoped to the frozen-foundation-encoder regime of Rein and VFM-UDA, and within it the tail gain holds on GTA→Cityscapes and not on DarkZurich (App. A).

5 Conclusion

Scene adaptation trades domain alignment against label preservation, and existing methods, which regulate how far pixel features move and leave the direction free, settle this trade-off at the expense of the small categories. In a hyperbolic embedding the label is angular while the shift is only partly radial, so a transport's *angular action* is the quantity that measures and controls the trade-off. `RadDA` turns that scalar into a margin, a label-validity and a step-count guarantee, and with a frozen foundation encoder on GTA→Cityscapes it wins on the tail where self-training does not; on DarkZurich and on encoders already adapted by self-training it does not. Because ε is measurable for any transport, the diagnosis outlives the method (App. A). Two directions follow. The optimum of the budget depends on the domain pair, so reading it off the data, for instance from the measured radial fraction of the shift, would remove the one hyper-parameter still tuned on the reported sets; and because a self-trained host has little angular gap left, applying the budget while self-training is still under way, rather than to a host that has already converged, is the natural next test, since that is where the angular gap is still open.

USE OF LARGE LANGUAGE MODELS

Generative AI tools were used substantially for drafting and revising the text of this paper, including parts of the introduction, the method section, the appendix and the figure captions, and for figure and table typesetting. They were not used to generate research ideas, to design or run experiments, to produce any experimental result, or to derive the proofs, all of which are the authors' own. Every AI-assisted passage was reviewed and revised by the authors, who take full responsibility for the content.

ETHICS STATEMENT

This work uses only publicly available driving-scene datasets under their licenses. Improved robustness of segmentation under adverse conditions has safety benefits for autonomous systems; we are not aware of specific harms introduced by the method beyond those inherent to the datasets.

REPRODUCIBILITY STATEMENT

All proofs are in App. C; the analysis protocol (embedding, radial/angular decomposition, computation of ε and ρ) is in App. D; architecture, hyper-parameters, pseudo-code and the exact host configurations are in App. E. Code and the trained heads will be released.

REFERENCES

Tomer Amit, Tal Shaharbany, Eliya Nachmani, and Lior Wolf. Segdiff: Image segmentation with diffusion probabilistic models. In *arXiv preprint*, 2021.

Nikita Araslanov and Stefan Roth. Self-supervised augmentation consistency for adapting semantic segmentation. In *CVPR*, 2021.

Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *ICLR*, 2022.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. In *Machine Learning*, 2010.

Yasser Benigmim, Subhankar Roy, Slim Essid, Vicky Kalogeiton, and Stéphane Lathuilière. One-shot unsupervised domain adaptation with personalized diffusion models. In *CVPR*, 2023.

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*, 2023a.

Ricky TQ Chen and Yaron Lipman. Flow matching on general geometries. In *ICLR*, 2024.

Ting Chen, Lala Li, Saurabh Saxena, Geoffrey Hinton, and David J Fleet. A generalist framework for panoptic segmentation of images and videos. In *ICCV*, 2023b.

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.

Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *NeurIPS*, 2017.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, 2013.

Dengxin Dai, Christos Sakaridis, Simon Hecker, and Luc Van Gool. Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding. In *IJCV*, 2020.

Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. Hyperbolic image-text representations. In *ICML*, 2023.

Brunó B. Englert and Gijs Dubbelman. VFM-UDA++: Improving network architectures and data strategies for unsupervised domain adaptive semantic segmentation. In *ICCV Workshops*, 2025.

Aleksandr Ermolov, Leyla Mirvakhabova, Valentin Khrulkov, Nicu Sebe, and Ivan Oseledets. Hyperbolic vision transformers: Combining improvements in metric learning. In *CVPR*, 2022.

Luca Franco, Paolo Mandica, Konstantinos Kallidromitis, Devin Guillory, Yu-Teng Li, Trevor Darrell, and Fabio Galasso. Hyperbolic active learning for semantic segmentation under domain shift. In *ICML*, 2024.

Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. In *NeurIPS*, 2018.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. In *JMLR*, 2016.

Jin Gao, Jialing Zhang, Xihui Liu, Trevor Darrell, Evan Shelhamer, and Dequan Wang. Back to the source: Diffusion-driven adaptation to test-time corruptions. In *CVPR*, 2023.

Mina Ghadimi Atigh, Martin Keller-Ressel, and Pascal Mettes. Hyperbolic busemann learning with ideal prototypes. In *NeurIPS*, 2021.

Mina Ghadimi Atigh, Julian Schoep, Erman Acar, Nanne Van Noord, and Pascal Mettes. Hyperbolic image segmentation. In *CVPR*, 2022.

Yaqian Guo, Xin Wang, Ce Li, and Shihui Ying. Domain adaptive semantic segmentation by optimal transport. In *arXiv preprint*, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

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.

Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, 2022a.

Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Hrda: Context-aware high-resolution domain-adaptive semantic segmentation. In *ECCV*, 2022b.

Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. Mic: Masked image consistency for context-enhanced domain adaptation. In *CVPR*, 2023.

Yuanfeng Ji, Zhe Chen, Enze Xie, Lanqing Hong, Xihui Liu, Zhaoqiang Liu, Tong Lu, Zhenguo Li, and Ping Luo. Ddp: Diffusion model for dense visual prediction. In *ICCV*, 2023.

Yuru Jia, Lukas Hoyer, Shengyu Huang, Tianfu Wang, Luc Van Gool, Konrad Schindler, and Anton Obukhov. Dginstyle: Domain-generalizable semantic segmentation with image diffusion models and stylized semantic control. In *ECCV*, 2024.

Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *CVPR*, 2020.

Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In *ICML*, 2020.

Georges Le Bellier and Nicolas Audebert. Floweo: Generative unsupervised domain adaptation for earth observation. In *WACV*, 2026.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2023.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023.

Gal Mishne, Zhengchao Wan, Yusu Wang, and Sheng Yang. The numerical stability of hyperbolic representation learning. In *ICML*, 2023.

Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *NeurIPS*, 2017.

Joshua Niemeijer, Manuel Schwonberg, Jan-Aike Termöhlen, Nico M Schmidt, and Tim Fingscheidt. Generalization by adaptation: Diffusion-based domain extension for domain-generalized semantic segmentation. In *WACV*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. In *TMLR*, 2024.

Duo Peng, Ping Hu, Qiuhong Ke, and Jun Liu. Diffusion-based image translation with label guidance for domain adaptive semantic segmentation. In *ICCV*, 2023.

Aram-Alexandre Pooladian, Heli Ben-Hamu, Carles Domingo-Enrich, Brandon Amos, Yaron Lipman, and Ricky TQ Chen. Multisample flow matching: Straightening flows with minibatch couplings. In *ICML*, 2023.

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

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *ICCV*, 2019.

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

Junjiao Tian, Lavisha Aggarwal, Andrea Colaco, Zsolt Kira, and Mar Gonzalez-Franco. Diffuse, attend, and segment: Unsupervised zero-shot segmentation using stable diffusion. In *CVPR*, 2024.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. In *arXiv preprint*, 2024.

Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *WACV*, 2021.

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.

VERIFY-AUTHORS. Flow matching adaptation for few-shot vision-language alignment. In *arXiv preprint*, 2025a.

VERIFY-AUTHORS. Raft: A domain adaptation framework for rgb & lidar semantic segmentation. In *arXiv preprint*, 2025b.

VERIFY-AUTHORS. Symmadapt: Symmetrical flow matching for source-free domain adaptation in medical image segmentation. In *arXiv preprint*, 2026a.

VERIFY-AUTHORS. Semantic alignment in hyperbolic space for open-vocabulary semantic segmentation. In *arXiv preprint*, 2026b.

VERIFY-AUTHORS. Path-decoupled hyperbolic flow matching for few-shot adaptation. In *arXiv preprint*, 2026c.

Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*, 2019.

Zhixiang Wei, Lin Chen, Yi Jin, Xiaoxiao Ma, Tianle Liu, Pengyang Ling, Ben Wang, Huaian Chen, and Jinjin Zheng. Stronger, fewer, & superior: Harnessing vision foundation models for domain generalized semantic segmentation. In *CVPR*, 2024.

Weijia Wu, Yuzhong Zhao, Mike Zheng Shou, Hong Zhou, and Chunhua Shen. Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models. In *ICCV*, 2023.

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.

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.

Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *CVPR*, 2023.

Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *CVPR*, 2020.

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 *CVPR*, 2021.

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