Hyperbolic Active Learning for Semantic Segmentation under Domain Shift

Luca Franco [*1] **Paolo Mandica** [*2] **Konstantinos Kallidromitis** [3] **Devin Guillory** [4] **Yu-Teng Li** [4] **Trevor Darrell** [4] **Fabio Galasso** [1 2]

Abstract

We introduce a hyperbolic neural network approach to pixel-level active learning for semantic segmentation. Analysis of the data statistics leads to a novel interpretation of the hyperbolic radius as an indicator of data scarcity. In HALO (Hyperbolic Active Learning Optimization), for the first time, we propose the use of epistemic uncertainty as a data acquisition strategy, following the intuition of selecting data points that are the least known. The hyperbolic radius, complemented by the widely-adopted prediction entropy, effectively approximates epistemic uncertainty. We perform extensive experimental analysis based on two established synthetic-to-real benchmarks, i.e. GTAV $\rightarrow$ Cityscapes and SYNTHIA $\rightarrow$ Cityscapes. Additionally, we test HALO on Cityscape $\rightarrow$ ACDC for domain adaptation under adverse weather conditions, and we benchmark both convolutional and attention-based backbones. HALO sets a new state-of-the-art in active learning for semantic segmentation under domain shift and it is the first active learning approach that surpasses the performance of supervised domain adaptation while using only a small portion of labels (i.e., 1%).[1]

1. Introduction

Dense prediction tasks, such as semantic segmentation (SS), are important in applications such as self-driving cars, manufacturing, and medicine. However, these tasks necessitate pixel-wise annotations, which can incur substantial costs and time inefficiencies (Cordts et al., 2016). Previous methods (Xie et al., 2022a; Vu et al., 2019; Shin et al., 2021b;a; Ning et al., 2021) have addressed this labeling challenge via domain adaptation, capitalizing on large source datasets for pre-training and domain-adapting with few target annotations (Ben-David et al., 2010). Most recently, active domain adaptation (ADA) has emerged as an effective strategy, i.e. annotating only a small set of target pixels in successive labelling rounds (Ning et al., 2021). State-of-the-art (SOTA) ADA (Shin et al., 2021b; Wu et al., 2022; Xie et al., 2022a) relies on the entropy of predicted pseudo-labels, which they define as prediction uncertainty, as the core strategy for active learning (AL) data acquisition. In fact, prediction uncertainty correlates well with the likelihood of pixel classification mistakes, but it is only one of the factors of the overall model uncertainty, as we argue in this work.

Following extensive literature (Depeweg et al., 2017; Kendall & Gal, 2017; Hüllermeier & Waegeman, 2021; Valdenegro-Toro & Mori, 2022), we distinguish aleatoric and epistemic uncertainty, and we propose the second for the data acquisition strategy in active learning. Epistemic uncertainty is an indicator of the state of knowledge about the task. This uncertainty stems not only from inaccuracies, as identified by prediction uncertainty, but also from the information the model has been exposed to thus far, including the amount of data considered. In the domain adaptation task, the domain gap arises from the model's lack of understanding of the new domain data, akin to the definition of epistemic uncertainty. Building upon this intuition, we propose HALO (Hyperbolic Active Learning Optimization), a novel approach for active domain adaptation, where we introduce the use of epistemic uncertainty into the data acquisition strategy. Our in-depth analysis shows that the hyperbolic radius effectively estimates data scarcity, revealing it as a key component in the estimation of epistemic uncertainty. The combination of the radius with a complementary information signal such as prediction entropy offers a comprehensive estimate of epistemic uncertainty.

Interpreting the hyperbolic radius as a proxy to data scarcity diverges from known interpretations in the hyperbolic literature. The SOTA hyperbolic SS model (Atigh et al., 2022) trains with class hierarchies, which they manually define. As a result, their hyperbolic radius represents the parent-to-child hierarchical relations in the Poincaré ball. We adopt Atigh et al. (2022) without enforcing hierarchi-

[*]Equal contribution [1]ITALAI S.R.L. [2]Sapienza University of Rome [3]Panasonic North America [4]UC Berkeley. Correspondence to: Luca Franco <luca.franco@italailabs.com>, Paolo Mandica <paolo.mandica@uniroma1.it>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

[1]Code available at https://github.com/paolomandica/HALO.

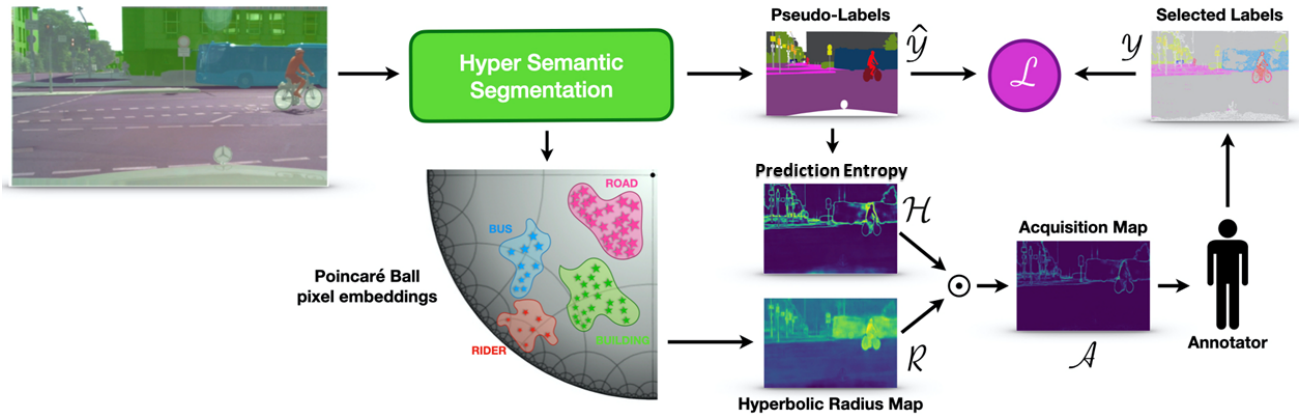

Figure 1. Overview of HALO. Pixels are encoded into the hyperbolic Poincaré ball and classified in the pseudo-label $\hat{\mathcal{Y}}$. The hyperbolic radius of the pixel embeddings defines the new hyperbolic score map $\mathcal{R}$. The prediction entropy $\mathcal{H}$ is extracted as the entropy of the softmax probabilities. Combining $\mathcal{R}$ and $\mathcal{H}$ we define the data acquisition score map $\mathcal{A}$, which is used to query new labels $\mathcal{Y}$.

cal labels and we find that hierarchical relationships do not emerge naturally in our case. For instance, in HALO, classes such as *road* and *building* are closer to the center of the ball, while *person* and *rider* have larger radii. This class arrangement contradicts the interpretation of the hyperbolic radius as a proxy for uncertainty, which emerged from metric learning hyperbolic studies (Ermolov et al., 2022; Franco et al., 2023). In our context, larger radii indicate larger data scarcity, therefore less certainty, which is in contrast with Franco et al. (2023)'s interpretation. Thus, our interpretation of the hyperbolic radius as a proxy for data scarcity does not align with neither of the existing interpretations in the case of hierarchy-free hyperbolic SS. Consider the HALO pipeline illustrated in Fig. 1 and the circular sector representing the Poincaré ball, where pixels from various classes are mapped. The hyperbolic model assigns a higher radius to classes that appear less frequently in the dataset (e.g., *rider*), and a lower radius to classes which are more frequent (e.g., *road*). In Sec. 4, we show how this novel interpretation of the hyperbolic radius arises bottom-up from data statistics.

We demonstrate the effectiveness of our approach through extensive benchmarking on well-established datasets for SS, including ADA from GTAV to Cityscapes, SYNTHIA to Cityscapes, and additional testing on Cityscapes to ACDC under adverse weather conditions. HALO sets a new SOTA on all the benchmarks (+3.3% on GTA→CS, +4.2% on SYNTHIA→CS, and +2.9% on CS→ACDC). Moreover, this is the first AL method that surpasses the supervised domain adaptation baseline using only a small portion of labels (+2.6% on GTA→CS with 5% budget). Our paper also introduces a novel technique to enhance the stability of hyperbolic training, referred to as *Hyperbolic Feature Reweighting* (HFR), cf. Sec. 5. Our code will be released.

In summary, our contributions include: 1) Presenting a novel interpretation of the hyperbolic radius as a proxy for data scarcity and its relationship with epistemic uncertainty; 2) Introducing hyperbolic neural networks in AL and a novel pixel-based data acquisition score based on the hyperbolic radius; 3) Validating both the concept and the algorithm through a comprehensive analysis, achieving a new state-of-the-art performance across all the considered ADA benchmarks for SS. Our method surpasses for the fist time in AL the supervised DA performance using only a small percentage of labels.

2. Related Works

Hyperbolic Representation Learning (HRL) Hyperbolic geometry has been extensively used to capture embeddings of tree-like structures (Nickel & Kiela, 2017; Chami et al., 2020) with low distortion (Sala et al., 2018; Sarkar, 2012). Since the seminal work of Ganea et al. (2018) on Hyperbolic Neural Networks (HNN), approaches have successfully combined hyperbolic geometry with model architectures ranging from convolutional (Shimizu et al., 2020) to attention-based (Gulcehre et al., 2018), including graph neural networks (Liu et al., 2019; Chami et al., 2019) and, most recently, vision transformers (Ermolov et al., 2022). There are two leading interpretations of the hyperbolic radius in hyperbolic space: as a measure of the prediction uncertainty (Chen et al., 2022; Ermolov et al., 2022; Franco et al., 2023; Flaborea et al., 2023) or as the hierarchical parent-to-child relation (Nickel & Kiela, 2017; Tifrea et al., 2018; Surís et al., 2021; Ermolov et al., 2022; Atigh et al., 2022). Our work builds on the SOTA hyperbolic semantic segmentation method of Atigh et al. (2022), which enforces hierarchical labels and training objectives. However, when training without manually injected hierarchical labels, as we do, the hierarchical interpretation does not apply. Although a correlation between the hyperbolic radius and an uncer-

tainty measure has been noted, a comprehensive understanding of this relationship is still lacking. In order to further research in this direction, we provide an investigation that examines the relationship between the hyperbolic radius, data scarcity, and epistemic uncertainty, aiming to shed light on this association. Furthermore, HALO's acquisition score is tailored for semantic segmentation, as it computes the hyperbolic radius for each pixel embedding. Hyperbolic neural networks have shown comparable performance to Euclidean models in semantic segmentation (Atigh et al., 2022), enabling fair comparisons. However, this equivalence does not extend to other tasks, where hyperbolic neural networks have not achieved similar performance (image classification) or are yet to be developed (object detection).

Active Learning (AL) The number of annotations required for dense tasks such as semantic segmentation can be costly and time-consuming. Active learning balances the labeling efforts and performance, selecting the most informative pixels in successive learning rounds. Strategies for active learning are based on uncertainty sampling (Gal et al., 2017; Wang & Shang, 2014; Wang et al., 2016), diversity sampling (Ash et al., 2019; Kirsch et al., 2019; Sener & Savarese, 2017; Wu et al., 2021) or a combination of both (Sinha et al., 2019; Xie et al., 2022b; Prabhu et al., 2021; Xie et al., 2022a). For the case of AL in semantic segmentation, EqualAL (Golestaneh & Kitani, 2020) incorporates the self-supervisory signal of self-consistency to mitigate the overfitting of scenarios with limited labeled training data. Labor (Shin et al., 2021b) selects the most representative pixels within the generation of an inconsistency mask. PixelPick (Shin et al., 2021a) prioritizes the identification of specific pixels or regions over labeling the entire image. Mittal et al. (2023) explores the effect of data distribution, semi-supervised learning, and labeling budgets. We are the first to leverage the hyperbolic radius as a proxy for the most informative pixels to label next.

Active Domain Adaptation (ADA) Domain Adaptation (DA) involves learning from a source data distribution and transferring that knowledge to a target dataset with a different distribution. Recent advancements in DA for semantic segmentation have utilized unsupervised (UDA) (Hoffman et al., 2018; Vu et al., 2019; Yang & Soatto, 2020; Liu et al., 2020; Mei et al., 2020; Liu et al., 2021) and semi-supervised (SSDA) (French et al., 2017; Saito et al., 2019; Singh, 2021; Jiang et al., 2020) learning techniques. However, challenges such as noise and label bias still pose limitations on the performance of DA methods. Active Domain Adaptation (ADA) aims to reduce the disparity between source and target domains by actively selecting informative data points from the target domain (Su et al., 2020; Fu et al., 2021; Singh et al., 2021; Shin et al., 2021b), which are subsequently labeled by human annotators. In semantic segmentation, Ning et al. (2021) propose a multi-anchor strategy to mitigate the distortion between the source and target distributions. The recent study of Xie et al. (2022a) shows the advantages of region-based selection in terms of region impurity and prediction uncertainty scores, compared to pixel-based approaches. By contrast, we show that selecting pixels just from class boundaries limits performance, as they are not necessarily the most informative, as we confirm with an oracular study. Instead, we show that the hyperbolic radius, in conjunction with prediction entropy, effectively approximates epistemic uncertainty, thereby serving as a successful objective for label acquisition.

Uncertainty The notion of uncertainty has gained increasing attention in machine learning (ML) research in recent years, primarily due to its growing practical significance in real-world applications. Consequently, numerous studies have developed approaches for uncertainty quantification in ML (Kendall & Gal, 2017; Carvalho et al., 2020; Xiao & Wang, 2019; Michelmore et al., 2020). Within the existing literature, two distinct sources of uncertainty are commonly acknowledged: aleatoric and epistemic (Fisher, 1930; Hora, 1996). Aleatoric uncertainty stems from the inherent randomness and variability within the data, while epistemic uncertainty arises from a lack of knowledge or data. As a result, epistemic uncertainty can theoretically be reduced with supplementary information, while aleatoric uncertainty remains non-reducible. Several methodologies have proposed techniques for quantifying both aleatoric and epistemic uncertainty (Depeweg et al., 2017; Kendall & Gal, 2017; Hüllermeier & Waegeman, 2021; Valdenegro-Toro & Mori, 2022). Following the approach of Depeweg et al. (2017), both total uncertainty and aleatoric uncertainty can be approximated via model ensemble (Lakshminarayanan et al., 2016), deriving epistemic uncertainty as the difference between the two. In our study, for the first time, we distinguish two leading complementary causes for epistemic uncertainty: prediction error and data scarcity.

3. Background

In this section, we introduce the background for our work. We begin by discussing Hyperbolic Neural Networks and Hyperbolic Semantic Segmentation, moving then to Active Domain Adaptation, which form the basis of our approach.

Hyperbolic Neural Networks and Semantic Segmentation We operate in the Poincaré ball hyperbolic manifold. We define it as the pair $(\mathbb{D}_c^N, g^{\mathbb{D}_c})$ where $\mathbb{D}_c^N = \{x \in \mathbb{R}^N : c\|x\| < 1\}$ is the manifold and $g_x^{\mathbb{D}_c} = (\lambda_x^c)^2 g^{\mathbb{E}}$ is the associated Riemannian metric, $-c$ is the curvature, $\lambda_x^c = \frac{2}{1-c\|x\|^2}$ is the conformal factor and $g^{\mathbb{E}} = \mathbb{I}^N$ is the Euclidean metric tensor. Hyperbolic neural networks first extract a feature vector v in Euclidean space, which is subsequently projected

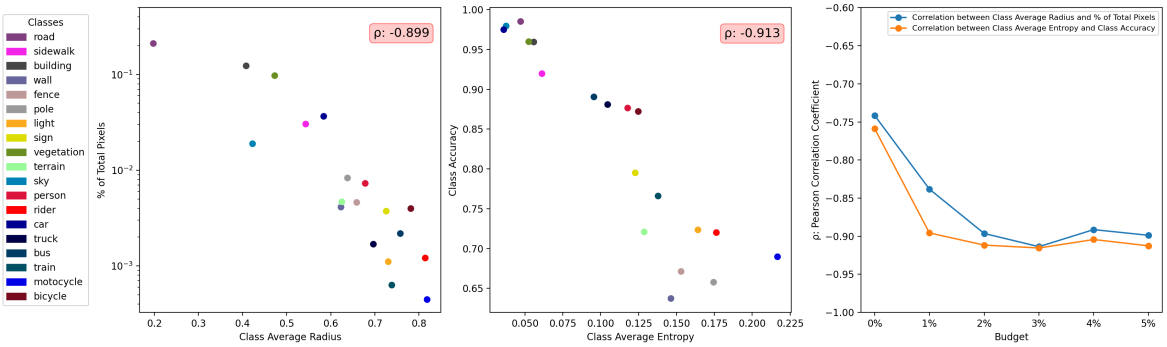

Figure 2. (left) Plot of class average radius vs. the percentage of total pixels in the target dataset; (center) Plot of the class average entropy vs. class accuracy; (right) Plot of labeling budget vs. correlation between class average radius and percentage of total pixels (blue) and between class average entropy and class accuracy (orange).

into the Poincaré ball via exponential map:

$$exp_x^c(v) = x \oplus_c \left(\frac{v}{\sqrt{c}\|v\|} tanh\left(\sqrt{c} \frac{\lambda_x^c \|v\|}{2} \right) \right) \quad (1)$$

where $x \in \mathbb{D}_c^N$ is the anchor and $\oplus_c$ is the Möbius hyperbolic addition. The latter is defined for two hyperbolic vectors h, w as follows:

$$h \oplus_c w = \frac{(1 + 2c\langle h, w\rangle + c\|w\|^2)v + (1 - c\|h\|^2)w}{1 + 2c\langle h, w\rangle + c^2\|h\|^2\|w\|^2} \quad (2)$$

We define the hyperbolic radius of the embedding $h \in \mathbb{D}_c^N$ as the Poincaré distance (See Eq. A1 in Appendix A.2) from the origin of the ball:

$$d(h, 0) = \frac{2}{\sqrt{c}} tanh^{-1}\left(\sqrt{c}\|h\|\right). \quad (3)$$

For the hyperbolic semantic segmentation, we adopt the work by Atigh et al. (2022), which stands as the first to showcase performance comparable to that of Euclidean networks. Segmentation is performed using hyperbolic multinomial logistic regression (HyperMLR) (Ganea et al., 2018). The complete formulation of HyperMLR is in Appendix A.2.

ADA for Semantic Segmentation The task aims to transfer knowledge from a source labeled dataset $\mathcal{S} = (X_s, Y_s)$ to a target unlabeled dataset $\mathcal{T} = (X_t, Y_t)$, where X represents an image and Y the corresponding annotation map. Y_s is given, Y_t is initially the empty set $\emptyset$. Adhering to the ADA protocol (Xie et al., 2022a; Wu et al., 2022; Shin et al., 2021b), target annotations are incrementally added in rounds, subject to a predefined budget, upon querying an annotator. Each pixel is assigned a priority score using a predefined acquisition map $\mathcal{A}$. Labels are added to Y_t in each AL round by selecting pixels from $\mathcal{A}$ with higher scores, in accordance with the budget. Each AL round is divided into two phases. In the first phase, the segmentation model undergoes end-to-end training, with back-propagation incorporating estimates $\hat{Y}_s$ and $\hat{Y}_t$ from the per-pixel cross-entropy loss $\mathcal{L}(\hat{Y}_s, \hat{Y}_t, Y_s, Y_t)$. The second phase consists in acquiring new target labels according to the acquisition score $\mathcal{A}$ and the predefined budget.

In Sec. 4, we assume to have pre-trained the hyperbolic image segmenter of Atigh et al. (2022) on the source dataset GTAV (Richter et al., 2016) and to have domain-adapted it to the target dataset Cityscapes (Cordts et al., 2016) through 5 rounds of AL with a total budget of 5%. The following analyses consider the radii of the hyperbolic pixel embeddings and the prediction entropy, for which statistics are computed on the Cityscapes validation set.

4. Hyperbolic Radius and Epistemic Uncertainty

In Sec. 4.1 we interpret the emerging properties of hyperbolic radius, and we compare with the interpretations in literature in Sec. 4.2.

4.1. Emerging properties of the hyperbolic radius

What does the hyperbolic radius represent? Fig. 2 (left) shows the correlation between the average class hyperbolic radius and the percentage of pixel labels for each class relative to the total number of pixels in the dataset. The correlation is substantial ($\rho = -0.899$), so classes with larger hyperbolic radii such as *motocycle* are rarer in the target dataset, while at lower hyperbolic radii we have more frequent classes such as *road*. In conclusion, larger hyperbolic radii indicate which classes the model has been exposed less so far in the training.

Understanding the role of the prediction entropy Prior active learning literature (Xie et al., 2022b; Prabhu et al., 2021; Xie et al., 2022a) agree on the utility of prediction entropy, i.e. the entropy of the prediction scores, in the

data acquisition strategy. In Fig. 2 (center) we report the correlation between the class average entropy and the class accuracy, whose resulting value is a strong correlation of $\rho = -0.913$. In conclusion, prediction entropy appears to be a good indicator for classes with low accuracy. In HALO, we combine prediction entropy with the newly proposed hyperbolic radius.

How does learning the hyperbolic manifold proceed? Fig. 2 (right) illustrates the evolution, across active learning rounds, of the correlation between prediction entropy and accuracy (orange), and the correlation between the hyperbolic radius and the percentage of pixels in the target dataset (blue). Both correlations exhibit a growing trend in module, eventually saturating at high values. In conclusion, as the training progresses, both the hyperbolic radius and the prediction entropy become better estimators for data scarcity and prediction error.

Relation with the epistemic uncertainty Following the work of Depeweg et al. (2017), we quantify the epistemic uncertainty as the difference between total and aleatoric uncertainty. The total uncertainty is estimated by computing the entropy of the predictive posterior distribution $U_t(\boldsymbol{x}) = \mathcal{H}[p(y|\boldsymbol{x})]$. This formulation encompasses the epistemic uncertainty regarding the network parameters $\boldsymbol{\theta}$. To compute it, we first measure the aleatoric uncertainty as $U_a(\boldsymbol{x}) = E_{p(\boldsymbol{\theta}|\mathcal{D})}\mathcal{H}[p(y|\boldsymbol{\theta}, \boldsymbol{x})]$ and then we derive the epistemic uncertainty as the difference $U_e(\boldsymbol{x}) = U_t(\boldsymbol{x}) - U_a(\boldsymbol{x})$. The model ensemble approach (Lakshminarayanan et al., 2016) offers an effective means to approximate the posterior distribution $p(\boldsymbol{\theta}|\mathcal{D})$ using a finite ensemble of models $\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_M$. We can approximate the total uncertainty as

$$\tilde{U}_t(\boldsymbol{x}) = -\sum_{y\in\mathcal{Y}} \left(\frac{1}{M}\sum_{m=1}^{M} p(y|\boldsymbol{\theta}_m, \boldsymbol{x})\right) log_2 \left(\frac{1}{M}\sum_{m=1}^{M} p(y|\boldsymbol{\theta}_m, \boldsymbol{x})\right) \tag{4}$$

and similarly the aleatoric uncertainty

$$\tilde{U}_a(\boldsymbol{x}) = -\frac{1}{M}\sum_{m=1}^{M}\sum_{y\in\mathcal{Y}} p(y|\boldsymbol{\theta}_m, \boldsymbol{x})\, log_2\, p(y|\boldsymbol{\theta}_m, \boldsymbol{x}). \tag{5}$$

Finally the epistemic uncertainty is approximated by the difference $\tilde{U}_e(\boldsymbol{x}) = \tilde{U}_t(\boldsymbol{x}) - \tilde{U}_a(\boldsymbol{x})$.

The correlation between the epistemic uncertainty and the hyperbolic radius results in a value of $\rho = 0.769$, while the correlation between the epistemic uncertainty and the prediction entropy is $\rho = 0.789$. Supported by the fact that the correlation between the hyperbolic radius and the prediction entropy is moderate ($\rho = 0.658$), we conclude that they encode complementary signals for the uncertainty description, respectively data scarcity and prediction error. In fact, the correlation between their product and epistemic uncertainty results in an even higher value ($\rho = 0.824$). Building upon this observation, we establish the acquisition score as the product of these two metrics (see Sec. 5.2).

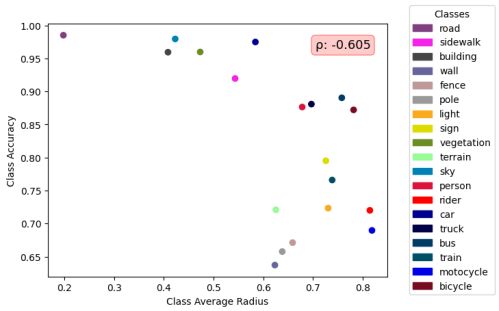

Figure 3. Plot of the class average radius vs. class accuracy.

4.2. Comparing interpretations of the hyperbolic radius

It emerges from our analysis that larger radii are assigned to classes that have higher data scarcity. Earlier work has explained the hyperbolic radius in terms of uncertainty or hierarchies. Approaches from the former (Franco et al., 2023; Flaborea et al., 2023) suggest that larger hyperbolic radii indicate more certain and unambiguous samples in terms of classification accuracy. In our case, the correlation between the hyperbolic radius and class accuracy, as depicted in Fig. 3, is moderate ($\rho = -0.605$). However, this value is considerably lower than the correlation between the hyperbolic radius and the percentage of pixels. Hence, the radius serves as a more effective indicator of data scarcity (see appendix A.7 for additional analysis). Another difference with the studies in favor of the uncertainty interpretation lies in the definition of the uncertainty as $1 -$ radius, typical of hyperbolic metric learning-based approaches (Franco et al., 2023; Flaborea et al., 2023). In those, a larger radius leads to an exponentially larger matching penalty due to the employed Poincaré distance, effectively making the radius inversely proportional to the errors, as those studies show.

Elsewhere, the interpretation of the hyperbolic radius aligns with a hierarchical explanation (Nickel & Kiela, 2017; Tifrea et al., 2018; Surís et al., 2021; Ermolov et al., 2022; Atigh et al., 2022). These methods involve hierarchical datasets, hierarchical labeling, and classification objective functions. Hierarchies naturally align with the growing volume in the Poincaré ball, resulting in children nodes from different parents being mapped further from each other than from their parents. Learning under hierarchical constraints results in leaf classes closer to the edge of the ball, and transitions between them traverse their parent nodes at lower hyperbolic radii. Our hyperbolic segmentation approach differs from prior hyperbolic works (Atigh et al., 2022; Franco et al., 2023; Flaborea et al., 2023; Ermolov et al., 2022) as we employ hyperbolic multinomial logistic regression without the incorporation of hierarchical labels or losses based on the Poincaré distance. These differences drive our intuition to utilize the hyperbolic radius as an estimator of data scarcity, thereby incorporating it into the final data acquisition score in the active learning process.

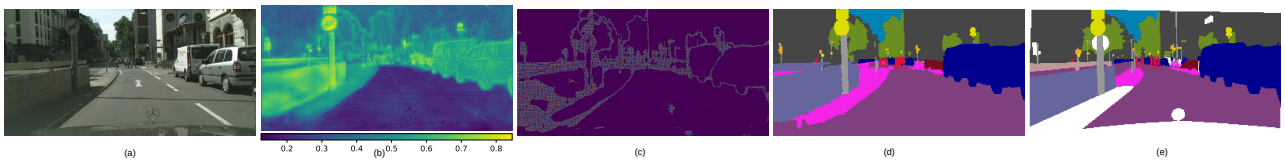

Figure 4. (a) Original image; (b) Radius map depicting the hyperbolic radii of pixel embeddings; (c) Pixels (yellow) that have been selected for data acquisition. See Sec. 5 for details; (d) HALO prediction; (e) Ground Truth annotations. Zoom in for the details.

5. Hyperbolic Active Learning Optimization (HALO)

In this section, first we introduce the proposed HALO pipeline (Sec. 5.1), then we detail the novel acquisition strategy (Sec. 5.2). Finally, we present our proposition for fixing the hyperbolic training instability (Sec. 5.3).

5.1. HALO pipeline

Let us consider Fig. 1. Our AL strategy consists in assigning an acquisition score $\mathcal{A}$ to each pixel, based on the combination of hyperbolic radius and prediction entropy. We estimate the hyperbolic radius $\mathcal{R}$ from pixel embeddings (as detailed in Sec. 4 and illustrated in Fig. 4b). Concurrently, predicted classification probabilities are used to compute the prediction entropy $\mathcal{H}$, inspired by prior works (Paul et al., 2020; Shin et al., 2021a; Wang & Shang, 2014; Wang et al., 2016; Xie et al., 2022a). New labels are subsequently chosen based on the acquisition score $\mathcal{A}$ and integrated into the training set (see selected pixels in Fig. 4c). Note that the new labels are both at the boundaries and within, in areas with the largest inaccuracies (compare Fig. 4d and 4e).

5.2. Novel data acquisition strategy

The acquisition score of each pixel in an image is formulated as the element-wise multiplication of the hyperbolic radius map $\mathcal{R}$ and the prediction entropy map $\mathcal{H}$, i.e. $\mathcal{A} = \mathcal{R} \odot \mathcal{H}$. The radius is computed as the distance of the hyperbolic pixel embedding (i, j) from the center of the Poincaré ball $\mathcal{R}^{(i,j)} = d(h_{i,j}, 0)$ (see Eq. 3). The prediction entropy $\mathcal{H}^{(i,j)} = -\sum_{c=1}^{C} P_{i,j,c}\, log P_{i,j,c}$ is estimated as the entropy of the softmax probability array $P_{i,j,c}$ associated with the pixel (i, j) and the classes $c \in \{1, ..., C\}$. The acquisition score $\mathcal{A}$ serves as a surrogate indicator for the epistemic uncertainty of each pixel and determines which ones are presented to the human annotator for labeling, to augment the target label set Y_t.

5.3. Robust hyperbolic learning with feature reweighting

HNNs are prone to instability during training because of the unique topology of the Poincaré ball. More precisely, when embeddings approach the boundary, the occurrence of vanishing gradients can impede the learning process. Several solutions have been proposed in the literature to address this problem. Guo et al. (2022b) achieves robustness by clipping the largest values of the radii, Franco et al. (2023) makes it by curriculum learning, and van Spengler et al. (2023) needs to carefully initialize the hyperbolic network parameters. However, these approaches yield suboptimal performances or are not compatible with our use case (see Appendix A.1). Therefore, we introduce the *Hyperbolic Feature Reweighting (HFR)* module, designed to enhance training stability by reweighting features, prior to their projection onto the Poincaré ball. Given the feature map $Z \in \mathbb{R}^{\tilde{H} \times \tilde{W}}$ generated by the encoder, we compute the weights $L = \text{HFR}(Z) \in \mathbb{R}^{\tilde{H} \times \tilde{W}}$ and use them to rescale each entry of the normalized feature map, yielding $\tilde{Z} = \frac{Z}{|Z|} \odot L$, where $|Z| = \sum_{k=1}^{\tilde{H}\tilde{W}} z_{ij}$ and $\odot$ denotes the element-wise multiplication. Intuitively, reweighting prevents embeddings from getting too close to the boundaries, where the distances tend to infinity. Our proposed HFR module is end-to-end trained and it enables the model to dynamically adapt through the various stages of training, improving its robustness.

6. Results

In this section, we describe the benchmarks and we perform a comparative evaluation against the SOTA (Sec. 6.1). We conduct ablation studies on the components of HALO and additional analyses in Sec. 6.2 and 6.3. The implementation follows Xie et al. (2022a) (details in Appendix A.3).

Datasets For pre-training, we utilize GTAV (Richter et al., 2016) and SYNTHIA (Ros et al., 2016) synthetic datasets, each comprising 24,966 and 9,000 densely annotated images, with 19 and 16 classes, respectively. For ADA training and evaluation, we employ real-world target datasets, specifically **Cityscapes** (CS) train and val sets or **ACDC** train and test sets, both categorized into the same 19 classes. CS (Cordts et al., 2016) consists of 2,975 training samples and 500 validation samples. **ACDC** (Sakaridis et al., 2021) comprises 4,006 images captured under adverse conditions (fog, nighttime, rain, snow) to maximize the complexity and diversity of the scenes.

Table 1. Comparison of mIoU results for different methods on the (a) **GTAV→Cityscapes**, (b) **SYNTHIA→Cityscapes**, and (C) **Cityscapes→ACDC** benchmarks. Methods marked with ♯ are based on DeepLab-v3+ (Chen et al., 2018b), the ones with † on SegFormer-B4 (Xie et al., 2021), whereas all the others use DeepLab-v2 (Chen et al., 2018a).

Method	Budget	road	side.	buil.	wall	fence	pole	light	sign	veg.	terr.	sky	pers.	rider	car	truck	bus	train	motor.	bike	mIoU	mIoU*
(a) GTAV → Cityscapes																						
LabOR (Shin et al., 2021b)	2.2%	96.6	77.0	89.6	47.8	50.7	**48.0**	**56.6**	63.5	89.5	57.8	91.6	72.0	47.3	91.7	62.1	61.9	48.9	47.9	65.3	66.6	-
RIPU (Xie et al., 2022a)	2.2%	96.5	74.1	89.7	53.1	51.0	43.8	53.4	62.2	90.0	57.6	92.6	73.0	53.0	**92.8**	73.8	**78.5**	62.0	**55.6**	70.0	69.6	-
HALO (ours)	2.2%	**97.5**	**79.9**	**90.2**	**55.6**	**51.5**	45.3	56.2	**66.2**	**90.2**	**58.6**	**92.8**	**73.3**	**53.5**	92.6	**76.9**	76.2	**64.2**	55.2	**70.1**	**70.8**	-
AADA♯ (Su et al., 2020)	5%	92.2	59.9	87.3	36.4	45.7	46.1	50.6	59.5	88.3	44.0	90.2	69.7	38.2	90.0	55.3	45.1	32.0	32.6	62.9	59.3	-
MADA♯ (Ning et al., 2021)	5%	95.1	69.8	88.5	43.3	48.7	45.7	53.3	59.2	89.1	46.7	91.5	73.9	50.1	91.2	60.6	56.9	48.4	51.6	68.7	64.9	-
D^2ADA♯ (Wu et al., 2022)	5%	97.0	77.8	90.0	46.0	55.0	52.7	58.7	65.8	90.4	58.9	92.1	75.7	54.4	92.3	69.0	78.0	68.5	59.1	72.3	71.3	-
RIPU♯ (Xie et al., 2022a)	5%	97.0	77.3	90.4	**54.6**	53.2	47.7	55.9	64.1	90.2	59.2	93.2	75.0	54.8	92.7	73.0	79.7	68.9	55.5	70.3	71.2	-
HALO♯ (ours)	5%	**97.6**	**81.0**	**91.4**	53.7	**54.9**	**56.7**	**62.9**	**72.1**	**91.4**	**60.5**	**94.1**	**78.0**	**57.3**	**94.0**	**81.4**	**84.7**	**70.1**	**60.0**	**73.3**	**74.5**	-
HALO† (ours)	5%	**98.2**	**85.4**	**92.5**	**62.5**	**61.6**	**58.3**	**67.7**	**74.9**	**92.2**	**65.1**	**94.7**	**79.9**	**60.8**	**94.6**	**84.1**	**85.4**	**83.6**	**61.2**	**75.5**	**77.8**	-
Eucl. Supervised DA♯	100%	97.4	77.9	91.1	54.9	53.7	51.9	57.9	64.7	91.1	57.8	93.2	74.7	54.8	93.6	76.4	79.3	67.8	55.6	71.3	71.9	-
Hyper. Supervised DA♯	100%	97.6	81.2	90.7	49.9	53.2	53.5	58.0	67.2	91.0	59.1	93.9	74.2	52.6	93.1	76.4	81.0	67.0	55.0	70.8	71.9	-
(b) SYNTHIA → Cityscapes																						
RIPU (Xie et al., 2022a)	2.2%	96.8	76.6	89.6	45.0	47.7	45.0	53.0	62.5	90.6	-	**92.7**	73.0	52.9	93.1	-	80.5	-	52.4	70.1	70.1	75.7
HALO (ours)	2.2%	**97.5**	**81.7**	**90.5**	**52.8**	**52.8**	**45.6**	**57.3**	**67.1**	**91.2**	-	92.6	**74.5**	**54.9**	**93.3**	-	**81.6**	-	**55.2**	**71.1**	**72.5**	**77.6**
AADA♯ (Su et al., 2020)	5%	91.3	57.6	86.9	37.6	48.3	45.0	50.4	58.5	88.2	-	90.3	69.4	37.9	89.9	-	44.5	-	32.8	62.5	61.9	66.2
MADA♯ (Ning et al., 2021)	5%	96.5	74.6	88.8	45.9	43.8	46.7	52.4	60.5	89.7	-	92.2	74.1	51.2	90.9	-	60.3	-	52.4	69.4	68.1	73.3
D^2ADA♯ (Wu et al., 2022)	5%	96.7	76.8	90.3	48.7	51.1	54.2	58.3	68.0	90.4	-	93.4	77.4	56.4	92.5	-	77.5	-	58.9	73.3	72.7	77.7
RIPU♯ (Xie et al., 2022a)	5%	97.0	78.9	89.9	47.2	50.7	48.5	55.2	63.9	91.1	-	93.0	74.4	54.1	92.9	-	79.9	-	55.3	71.0	71.4	76.7
HALO♯ (ours)	5%	**97.5**	**81.5**	**91.5**	**56.5**	**52.7**	**57.0**	**63.2**	**72.9**	**92.0**	-	**94.4**	**77.8**	**57.4**	**94.4**	-	**86.1**	-	**60.5**	**73.5**	**75.6**	**80.2**
HALO† (ours)	5%	**98.3**	**86.5**	**92.6**	**61.0**	**61.5**	**60.6**	**67.6**	**76.2**	**93.2**	-	**94.6**	**80.8**	**58.9**	**95.0**	-	85.1	-	**62.7**	**75.6**	**78.1**	**82.1**
Eucl. Supervised DA♯	100%	97.5	81.4	90.9	48.5	51.3	53.6	59.4	68.1	91.7	-	93.4	75.6	51.9	93.2	-	75.6	-	52.0	71.2	72.2	77.1
Hyper. Supervised DA♯	100%	97.7	82.2	90.3	53.0	48.8	51.7	56.0	66.1	91.4	-	94.2	75.0	51.5	93.4	-	82.1	-	52.8	70.2	72.3	77.1
(c) Cityscapes → ACDC																						
RIPU (Xie et al., 2022a)	2.2%	91.4	69.5	83.8	**52.7**	41.6	52.8	66.4	54.2	**85.1**	47.5	**94.7**	54.5	21.8	**85.5**	**58.7**	58.8	**76.9**	41.4	**45.9**	62.3	-
HALO	2.2%	**92.6**	**71.3**	**84.5**	51.3	**43.1**	**53.5**	**67.2**	**57.6**	85.1	**49.5**	94.5	**57.2**	**28.6**	84.1	53.3	**76.0**	66.9	**44.1**	41.4	**63.2**	-
RIPU♯ (Xie et al., 2022a)	5%	**92.7**	**72.5**	84.7	53.1	44.8	56.7	69.1	58.9	85.9	46.9	**95.3**	57.2	24.3	84.5	**61.4**	59.4	79.0	36.9	43.6	63.5	-
HALO♯	5%	92.6	72.2	**84.8**	**54.9**	**47.7**	**59.5**	**71.5**	**61.1**	**86.1**	**49.5**	95.2	**60.7**	**30.6**	**85.8**	58.4	**73.8**	**82.0**	**41.6**	**53.2**	**66.4**	-
HALO†	5%	**95.2**	**79.8**	**88.2**	**60.2**	**51.1**	**64.1**	**78.2**	**65.6**	**87.9**	**55.7**	**95.5**	**66.3**	20.7	**88.9**	**82.2**	**89.3**	**87.9**	**50.4**	**59.0**	**71.9**	-

Training protocol The model undergoes a pre-training on either GTAV or SYNTHIA source synthetic datasets. Subsequently, the model is domain adapted using both the source and the target datasets. Our hyperbolic radius-based acquisition method is used to select pixels to be labeled in five evenly spaced rounds during training, with either 2.2% or 5% of the total labels. Our model is additionally trained under adverse weather conditions, using CS and ACDC as the source and target datasets, respectively, in line with Hoyer et al. (2023) and Brüggemann et al. (2023). The ADA performances in Table 1 are also compared with the corresponding supervised domain adaptation baselines (Supervised DA). Supervised DA refers to the process where the adaptation to a target dataset involves using all of its labels (i.e., 100%) for the whole training, in contrast to active domain adaptation which uses a smaller fraction (e.g., 2.2% or 5%) of labels.

Evaluation metrics To assess the effectiveness of the models, the mean Intersection-over-Union (mIoU) metric is computed on the target validation set. For GTAV→CS and CS→ACDC, the mIoU is calculated on the shared 19 classes, whereas for SYNTHIA→CS two mIoU values are reported, one on the 13 common classes (mIoU*) and another on the 16 common classes (mIoU).

6.1. Comparison with the state-of-the-art

In Table 1a, we present the results of our method and the most recent ADA approaches on the GTAV→CS benchmark. HALO outperforms the current state-of-the-art methods (Xie et al., 2022a; Wu et al., 2022) using both 2.2% (+1.2% mIoU) and 5% (+3.3% mIoU) of labeled pixels, reaching 70.8% and 74.5%, respectively. Additionally, our method is the first to surpass the supervised domain adaptation baseline (71.9%), even by a significant margin (+2.6%). A thorough analysis on the performance at increasing budgets is provided in Sec. 6.3. HALO achieves state-of-the-art also in the SYNTHIA→CS case (cf. Table 1b), where it improves by +2.4% and +4.2% using 2.2% and 5% of labels, reaching performances of 72.5% and 75.6%, respectively. Additionally, we train HALO with the SegFormer-B4 (Xie et al., 2021) segmenter to demonstrate the adaptability of our approach to different architectures. With SegFormer-B4, HALO improves by +3.3% in GTAV→CS and +2.5% in SYNTHIA→CS compared to HALO with DeepLab-v3+, using 5% of labels. Due to the absence of other ADA studies on CS→ACDC adaptation, we train RIPU (Xie et al., 2022a) as a baseline for comparison with our method. HALO improves over RIPU by +2.9% mIoU with a 5% budget, reaffirming the effectiveness of our approach on a novel dataset,

Table 2. Ablation study conducted with the Hyperbolic DeepLab-v3+ as backbone with 5% budget. Performance of prediction entropy and hyperbolic radius scores in isolation (a and b) and combined (c).

Ablative version	mIoU
(a) Prediction Entropy only ($\mathcal{H}$)	63.2
(b) Hyperbolic Radius only ($\mathcal{R}$)	64.1
(c) **HALO** ($\mathcal{R} \odot \mathcal{H}$)	**74.5**

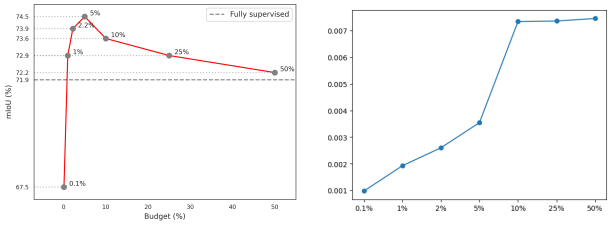

Figure 5. (left) Performance on GTAV → Cityscapes with different budgets. (right) Evolution of the variance (y axis) of selected pixels distributions with varying budget (x axis).

as shown in Table 1c.

6.2. Ablation study

We begin by conducting an oracular study using ground-truth labels, followed by ablation studies on the selection criteria, region- versus pixel-based acquisition scores, and HFR. Additional ablation studies are in Appendix A.1.

Oracle experiment with ground-truth boundaries To test our hypothesis that acquiring labels solely from class boundaries results in performance decline, we conduct an oracular experiment. We replace the pseudo-labels used in RIPU with ground-truth labels, effectively evaluating an AL acquisition strategy based on ground-truth boundary pixels. Although oracular, the experiment yields a performance drop of 1.4 mIoU (69.8 vs. to RIPU's 71.2), motivating the design of a novel acquisition strategy which samples also from non-boundary regions.

Selection criteria HALO demonstrates a substantial improvement of +10.4% compared to methods (a) and (b) in Table 2. More precisely, utilizing solely either the entropy (a) or the hyperbolic radius (b) as acquisition scores yields comparable performance of 63.2% and 64.1%, respectively. When these two metrics are combined, the final performance is notably improved to 74.5%.

Region- vs. Pixel-based criteria Unlike region impurity in Xie et al. (2022a), the hyperbolic radius is a continuous quantity that can be computed for each pixel. We conduct experiments comparing region- and pixel-based acquisition scores. The results demonstrate a small difference between the two approaches (74.1% vs. 74.5%), proving that HALO does not necessitate a region-based formulation. More in Appendix A.1.

Hyperbolic Feature Reweighting (HFR) HFR improves training stability and enhances performance in the Hyperbolic model. Although the mIoU improvement is modest (+1.6%), the main advantage is the training robustness, as the Hyperbolic model otherwise struggles to converge. HFR does not benefit the Euclidean model and instead negatively impacts its performance. More in Appendix A.1.

6.3. Additional analyses

Correlation analysis on Cityscapes→ACDC In addition to GTAV→CS, we report the correlations for CS→ACDC. The correlation of hyperbolic radius vs. percentage of target pixels is -0.868, while the correlation of prediction entropy vs. class accuracy is -0.892. These results are in line with the GTAV→CS case (-0.899 and -0.913), showing that the proposed method generalizes well even on a different domain adaptation benchmark.

Class imbalance with increasing budget We experiment with different labeling budgets, observing performance improvements as the number of labeled pixels increases. However, beyond a threshold of 5%, adding more labeled pixels leads to diminishing returns (see Fig. 5 (left)). We believe this may be explained by data unbalance: taking all labels to domain adapt means that most of them belong to a few classes. Indeed, *road*, *building* and *vegetation* account for 77% of the total labels, potentially hindering successive training rounds due to data redundancy.

To verify the intuition on data imbalance, we have evaluated the variance of selected pixels distributions as the labelling budget increases. In Fig. 5 (right), we start with the acquisition of just 0.1% of labels from the target dataset. At this stage, the variance is at a minimum, as HALO manages to identify and select labels from each class in equal proportions. Then the variance increases slowly until the budget reaches 5%. This happens as the model manages to select pixels from each class, balancing the acquired data selection. The variance has a steep increase at budgets of 10% and higher. This occurs because the model has already selected most of the labels from the complex and scarce classes which it can identify thanks to the hyperbolic radius and the prediction entropy (cf. Sec. 5.2). So, for budgets of 10% or more, the data acquisition strategy is influenced by the target dataset imbalance. The imbalance trend in label selection matches the performance variation in Fig. 5 (left). Therefore, we conclude that HALO's selection aids performance, beyond the supervised domain adaptation, until the model manages to successfully identify complex and scarce classes, and until they are available in the target dataset.

7. Conclusions

We have introduced the first hyperbolic neural network technique for AL, which we have extensively validated as the novel state-of-the-art on semantic segmentation under domain shift. We have identified a novel interpretation of the hyperbolic radius as an estimator of data scarcity and epistemic uncertainty, and we have supported the finding with experimental evidence. The novel concept of hyperbolic radius and its successful use as an acquisition strategy in AL are a step forward in understanding hyperbolic neural networks.

Limitations

While we have presented experimental evidence supporting the need for a novel interpretation of the hyperbolic radius, our work lacks a rigorous mathematical validation of the properties of the hyperbolic radius within the given experimental setup. Future research should delve into this mathematical aspect to formalize and prove these properties.

HALO's reliance on a source model pretrained on synthetic data introduces challenges related to large-scale simulation efforts and the need for effective synthetic-to-real domain adaptation. Exploring alternative strategies, such as self-supervised pre-training on real source datasets, could be a promising research direction to mitigate these challenges.

Although Active Domain Adaptation significantly reduces labeling costs, the manual annotation of individual pixels can be a time-consuming task. Further investigation into human-robot interaction methodologies to streamline pixel annotation processes and expedite the annotation workflow is needed.

Acknowledgements

We gratefully acknowledge Panasonic Corporation for funding this study. We also acknowledge financial support from the PNRR MUR project PE0000013-FAIR and from Regione Lazio (Italy), PO FSE 2014-2020 program.

Impact Statement

Hyperbolic Neural Networks (HNN) have recently gained prominence, achieving state-of-the-art performance across various tasks. However, the theory and interpretation of HNN remain diverse, particularly concerning the interpretation of the hyperbolic radius. While it has been traditionally viewed as a continuum hierarchical parent-to-child measure or as an estimate of uncertainty, our novel interpretation adds a new dimension to the growing framework of HNN, advancing the field further.

While there are many potential societal consequences of our work, we feel none must be specifically highlighted here.

References

Ash, J. T., Zhang, C., Krishnamurthy, A., Langford, J., and Agarwal, A. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

Atigh, M., Schoep, J., Acar, E., Noord, N. V., and Mettes, P. Hyperbolic image segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4443–4452, Los Alamitos, CA, USA, jun 2022. IEEE Computer Society.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2010.

Brüggemann, D., Sakaridis, C., Truong, P., and Van Gool, L. Refign: Align and refine for adaptation of semantic segmentation to adverse conditions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3174–3184, 2023.

Carvalho, E. D. C., Clark, R., Nicastro, A., and Kelly, P. H. J. Scalable uncertainty for computer vision with functional variational inference. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12000–12010, 2020. doi: 10.1109/CVPR42600.2020.01202.

Chami, I., Ying, Z., Ré, C., and Leskovec, J. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

Chami, I., Gu, A., Chatziafratis, V., and Ré, C. From trees to continuous embeddings and back: Hyperbolic hierarchical clustering. *Advances in Neural Information Processing Systems*, 33:15065–15076, 2020.

Chen, B., Peng, W., Cao, X., and Röning, J. Hyperbolic uncertainty aware semantic segmentation. *arXiv preprint arXiv:2203.08881*, 2022.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, Apr 2018a.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Computer Vision – ECCV 2018*, pp. 833–851, Cham, 2018b.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B.

The cityscapes dataset for semantic urban scene understanding. *CoRR*, abs/1604.01685, 2016.

Depeweg, S., Hernández-Lobato, J. M., Doshi-Velez, F., and Udluft, S. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *International Conference on Machine Learning*, 2017.

Ermolov, A., Mirvakhabova, L., Khrulkov, V., Sebe, N., and Oseledets, I. Hyperbolic vision transformers: Combining improvements in metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7409–7419, 2022.

Fisher, R. A. Inverse probability. *Mathematical Proceedings of the Cambridge Philosophical Society*, 26(4):528–535, 1930. doi: 10.1017/S0305004100016297.

Flaborea, A., Prenkaj, B., Munjal, B., Sterpa, M., Aragona, D., Podo, L., and Galasso, F. Are we certain it's anomalous? In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2897–2907, Los Alamitos, CA, USA, jun 2023. IEEE Computer Society. doi: 10.1109/CVPRW59228.2023.00291.

Franco, L., Mandica, P., Munjal, B., and Galasso, F. Hyperbolic self-paced learning for self-supervised skeleton-based action representations. In *The Eleventh International Conference on Learning Representations*, 2023.

French, G., Mackiewicz, M., and Fisher, M. Self-ensembling for visual domain adaptation. *arXiv preprint arXiv:1706.05208*, 2017.

Fu, B., Cao, Z., Wang, J., and Long, M. Transferable query selection for active domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7272–7281, 2021.

Gal, Y., Islam, R., and Ghahramani, Z. Deep bayesian active learning with image data. In *International conference on machine learning*, pp. 1183–1192. PMLR, 2017.

Ganea, O., Bécigneul, G., and Hofmann, T. Hyperbolic neural networks. *Advances in neural information processing systems*, 31, 2018.

Golestaneh, S. A. and Kitani, K. M. Importance of self-consistency in active learning for semantic segmentation. *BMVC*, 2020.

Gulcehre, C., Denil, M., Malinowski, M., Razavi, A., Pascanu, R., Hermann, K. M., Battaglia, P., Bapst, V., Raposo, D., Santoro, A., et al. Hyperbolic attention networks. *arXiv preprint arXiv:1805.09786*, 2018.

Guo, Y., Wang, X., Chen, Y., and Yu, S. X. Clipped hyperbolic classifiers are super-hyperbolic classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11–20, 2022a.

Guo, Y., Wang, X., Chen, Y., and Yu, S. X. Clipped hyperbolic classifiers are super-hyperbolic classifiers. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–10, Los Alamitos, CA, USA, jun 2022b. IEEE Computer Society.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., and Darrell, T. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pp. 1989–1998. Pmlr, 2018.

Hora, S. C. Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering & System Safety*, 54:217–223, 1996. URL https://api.semanticscholar.org/CorpusID:111162869.

Hoyer, L., Dai, D., Wang, H., and Van Gool, L. Mic: Masked image consistency for context-enhanced domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11721–11732, 2023.

Hüllermeier, E. and Waegeman, W. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, 110, 03 2021. doi: 10.1007/s10994-021-05946-3.

Jiang, P., Wu, A., Han, Y., Shao, Y., Qi, M., and Li, B. Bidirectional adversarial training for semi-supervised domain adaptation. In *IJCAI*, pp. 934–940, 2020.

Kendall, A. and Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? In *Neural Information Processing Systems*, 2017.

Kirsch, A., van Amersfoort, J., and Gal, Y. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning, 2019.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Neural Information Processing Systems*, 2016. URL https://api.semanticscholar.org/CorpusID:6294674.

Liu, Q., Nickel, M., and Kiela, D. Hyperbolic graph neural networks. *Advances in neural information processing systems*, 32, 2019.

Liu, Y., Zhang, W., and Wang, J. Source-free domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1215–1224, 2021.

Liu, Z., Miao, Z., Pan, X., Zhan, X., Lin, D., Yu, S. X., and Gong, B. Open compound domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12406–12415, 2020.

Lou, A., Katsman, I., Jiang, Q., Belongie, S., Lim, S.-N., and Sa, C. D. Differentiating through the fréchet mean, 2021.

Mei, K., Zhu, C., Zou, J., and Zhang, S. Instance adaptive self-training for unsupervised domain adaptation. In *Computer Vision–ECCV 2020: 16th European Conference Proceedings, Part XXVI 16*, pp. 415–430, 2020.

Michelmore, R., Wicker, M., Laurenti, L., Cardelli, L., Gal, Y., and Kwiatkowska, M. Uncertainty quantification with statistical guarantees in end-to-end autonomous driving control. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7344–7350, 2020. doi: 10.1109/ICRA40945.2020.9196844.

Mittal, S., Niemeijer, J., Schäfer, J. P., and Brox, T. Best practices in active learning for semantic segmentation, 2023.

Nickel, M. and Kiela, D. Poincaré embeddings for learning hierarchical representations, 2017.

Ning, M., Lu, D., Wei, D., Bian, C., Yuan, C., Yu, S., Ma, K., and Zheng, Y. Multi-anchor active domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9112–9122, 2021.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. 2019.

Paul, S., Tsai, Y.-H., Schulter, S., Roy-Chowdhury, A. K., and Chandraker, M. Domain adaptive semantic segmentation using weak labels, 2020.

Prabhu, V., Chandrasekaran, A., Saenko, K., and Hoffman, J. Active domain adaptation via clustering uncertainty-weighted embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8505–8514, 2021.

Richter, S. R., Vineet, V., Roth, S., and Koltun, V. Playing for data: Ground truth from computer games. In *European Conference on Computer Vision (ECCV)*, volume 9906, pp. 102–118, 2016.

Ros, G., Sellart, L., Materzynska, J., Vazquez, D., and Lopez, A. M. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Saito, K., Kim, D., Sclaroff, S., Darrell, T., and Saenko, K. Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Sakaridis, C., Dai, D., and Van Gool, L. ACDC: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021.

Sala, F., De Sa, C., Gu, A., and Ré, C. Representation tradeoffs for hyperbolic embeddings. In *International conference on machine learning*, pp. 4460–4469. PMLR, 2018.

Sarkar, R. Low distortion delaunay embedding of trees in hyperbolic plane. In *Graph Drawing: 19th International Symposium, GD 2011, Revised Selected Papers 19*, pp. 355–366, 2012.

Sener, O. and Savarese, S. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

Shimizu, R., Mukuta, Y., and Harada, T. Hyperbolic neural networks++. *arXiv preprint arXiv:2006.08210*, 2020.

Shin, G., Xie, W., and Albanie, S. All you need are a few pixels: semantic segmentation with pixelpick. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1687–1697, 2021a.

Shin, I., Kim, D.-J., Cho, J. W., Woo, S., Park, K., and Kweon, I. S. Labor: Labeling only if required for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8588–8598, 2021b.

Singh, A. Clda: Contrastive learning for semi-supervised domain adaptation. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 5089–5101. Curran Associates, Inc., 2021.

Singh, A., Doraiswamy, N., Takamuku, S., Bhalerao, M., Dutta, T., Biswas, S., Chepuri, A., Vengatesan, B., and Natori, N. Improving semi-supervised domain adaptation using effective target selection and semantics. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2703–2712, 2021.

Sinha, S., Ebrahimi, S., and Darrell, T. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5972–5981, 2019.

Su, J.-C., Tsai, Y.-H., Sohn, K., Liu, B., Maji, S., and Chandraker, M. Active adversarial domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 739–748, 2020.

Surís, D., Liu, R., and Vondrick, C. Learning the predictability of the future. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 12607–12617, 2021.

Tifrea, A., Bécigneul, G., and Ganea, O.-E. Poincaré glove: Hyperbolic word embeddings. *arXiv preprint arXiv:1810.06546*, 2018.

Valdenegro-Toro, M. and Mori, D. A deeper look into aleatoric and epistemic uncertainty disentanglement. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1508–1516, Los Alamitos, CA, USA, jun 2022. IEEE Computer Society. doi: 10.1109/CVPRW56347.2022.00157.

van Spengler, M., Berkhout, E., and Mettes, P. Poincaré resnet, 2023.

Vu, T.-H., Jain, H., Bucher, M., Cord, M., and Perez, P. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Wang, D. and Shang, Y. A new active labeling method for deep learning. In *2014 International joint conference on neural networks (IJCNN)*, pp. 112–119. IEEE, 2014.

Wang, K., Zhang, D., Li, Y., Zhang, R., and Lin, L. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600, 2016.

Wu, T.-H., Liu, Y.-C., Huang, Y.-K., Lee, H.-Y., Su, H.-T., Huang, P.-C., and Hsu, W. H. Redal: Region-based and diversity-aware active learning for point cloud semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15510–15519, 2021.

Wu, T.-H., Liou, Y.-S., Yuan, S.-J., Lee, H.-Y., Chen, T.-I., Huang, K.-C., and Hsu, W. H. D 2 ada: Dynamic density-aware active domain adaptation for semantic segmentation. In *Computer Vision ECCV 2022 Proceedings, Part XXIX*, pp. 449–467. Springer, 2022.

Xiao, Y. and Wang, W. Quantifying uncertainties in natural language processing tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:7322–7329, 07 2019. doi: 10.1609/aaai.v33i01.33017322.

Xie, B., Yuan, L., Li, S., Liu, C. H., and Cheng, X. Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8068–8078, June 2022a.

Xie, B., Yuan, L., Li, S., Liu, C. H., Cheng, X., and Wang, G. Active learning for domain adaptation: An energy-based approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8708–8716, 2022b.

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Neural Information Processing Systems (NeurIPS)*, 2021.

Yang, Y. and Soatto, S. Fda: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Appendix

This appendix provides additional information and insights on the proposed Hyperbolic Active Learning Optimization (HALO) for semantic segmentation under domain shift.

This supplementary material is structured as follows:

A.1: Additional ablation studies presents additional ablation studies on the proposed Hyperbolic Feature Reweighting (HFR), strategies for stable training in hyperbolic space, region- vs. pixel-based acquisition score, evaluation on the source-free protocol and correlation between Riemannian variance and classification accuracy;

A.2 Additional hyperbolic formulas reports additional employed hyperbolic formulas;

A.3 Implementation details describes the training details adopted in the experiments;

A.4 Comparison of parameters count compares the number of parameters of HALO with the baseline model;

A.5 Computational resources consumption analyzes the computational cost of HALO compared to the baseline;

A.6 Qualitative results showcases representative qualitative results of HALO;

A.7 Data acquisition strategy: rounds of selection illustrates examples of pixel labeling selection and the priorities of the data acquisition strategy at each acquisition round;

A.8 Qualitative comparison with the baseline model illustrates a qualitative comparison of pixel acquisition between HALO and baseline model to prove the limitation of boundary-only selection;

A.1. Additional ablation studies

A.1.1. Results of HFR

Table A1 provides insights into the performance of hyperbolic and Euclidean models with and without Hyperbolic Feature Reweighting (HFR). In the case of HALO, the performance with and without HFR remains the same in the source-only setting. However, when applied to the source+target ADA scenario, HFR leads to an improvement of 1.2%. It should be noted that HFR also stabilizes the training of hyperbolic models. In fact, when not using HFR, training requires a warm-up schedule and, still, it does not converge in approximately 20% of the runs. HFR improves therefore performance for ADA and it is important for hyperbolic learning stability.

Table A1. **HFR Performance Comparison:** Evaluating the impact of Hyperbolic Feature Reweighting (HFR) on hyperbolic and Euclidean models in source-only and source+target protocols.

Encoder	Protocol	HFR	mIoU (%)
DeepLab-v3+	source-only	✗	36.3
DeepLab-v3+	source-only	✓	22.7
Hyper DeepLab-v3+	source-only	✗	39.0
Hyper DeepLab-v3+	source-only	✓	38.9
HALO	source+target	✗	72.9
HALO	**source+target**	✓	**74.5**

A.1.2. Exploring strategies for stable training in hyperbolic space

Here we conduct a more comprehensive evaluation of approaches aimed at stabilizing the training of hyperbolic neural networks. We test Guo et al. (2022a)'s Feature Clipping method in our framework for comparison with our HFR. As shown in the Table A2, while Feature Clipping works and produces better results than the baseline RIPU, it still falls short of our HFR method (-1.2%). Guo et al. (2022a) utilize Feature Clipping to prevent vanishing gradients during backpropagation. Despite its simplicity, this technique restricts the model's representational capacity by clipping features,

resulting in inferior performance compared to our HFR. We have not tested the curriculum learning of Franco et al. (2023) and the initialization approach of van Spengler et al. (2023), because their adaptation to the ADA task is not straightforward. The curriculum learning in Franco et al. (2023) is specifically tailored for metric learning scenarios involving a hyperbolic loss, enabling training in hyperbolic space by utilizing cosine distance for improved initialization, gradually transitioning to the Poincaré loss. Our method does not involve comparing embeddings and leveraging the Poincaré loss. Similarly, the initialization approach in van Spengler et al. (2023) is designed explicitly for fully hyperbolic ResNets, particularly hyperbolic convolutions. As we do not employ hyperbolic convolutional layers, their initialization approach is not immediately suitable for our model.

Table A2. Comparison of strategies for stable training in hyperbolic space

Method	mIoU (%)
Hyperbolic Feature Reweighting (ours)	**74.5**
Feature Clipping (Guo et al., 2022)	73.3
Initialization (van Spengler et al., 2023)	not compatible
Curriculum Learning (Franco et al., 2023)	not compatible

A.1.3. Region- vs. Pixel-based acquisition score

While the region impurity score of RIPU (Xie et al., 2022a) requires pixel regions to work, as the impurity is based on region statistics, the hyperbolic radius employed in HALO can be computed on both pixel and region bases. Here we train HALO with the region-based approach for comparison. As we observe in the Table A3, the region-based approach leads to a small difference of -0.4% on the GTAV→CS benchmark with 5% acquired labels, but still manages to achieve a significative improvement over the baseline (RIPU).

Table A3. Comparison of Region- vs. Pixel-based acquisition score.

Method	Region-based	Pixel-based	mIoU (%)
RIPU (Xie et al., 2022a)	✓		71.2
HALO (ours)	✓		74.1
HALO (ours)		✓	**74.5**

A.1.4. Source-free domain adaptation

In the source-free protocol, the model is pre-trained on the source dataset and domain-adapted using only the target dataset. In Table A4 we show the performance of HALO on the source-free GTAV → CS domain adaptation task. HALO surpasses the current best (Xie et al., 2022a) by +3% using 2.2% of labels.

Table A4. HALO performance on the **source-free protocol** on GTAV→CS, compared with the previous state-of-the-art approach. Methods marked with ♯ are based on DeepLab-v3+ (Chen et al., 2018b), whereas all the others use DeepLab-v2 (Chen et al., 2018a).

Method	**Budget**	**mIoU**
RIPU (Xie et al., 2022a)	2.2%	67.1
HALO (ours)	2.2%	**70.1**
HALO♯ (ours)	5%	**73.3**

A.1.5. Analysis on the Riemannian variance

Fig. 6 complements our analysis by plotting the class accuracies vs. the Riemannian variance (see Eq. A2 in Appendix A.2) of radii for each class. The latter generalizes the Euclidean variance, considering the increasing Poincaré ball density

at larger radii. The correlation between accuracy and Riemannian variance is noteworthy ($\rho = -0.811$), indicating that challenging classes, like *pole*, exhibit lower accuracy and larger Riemannian variance, occupying a greater volume in the space.

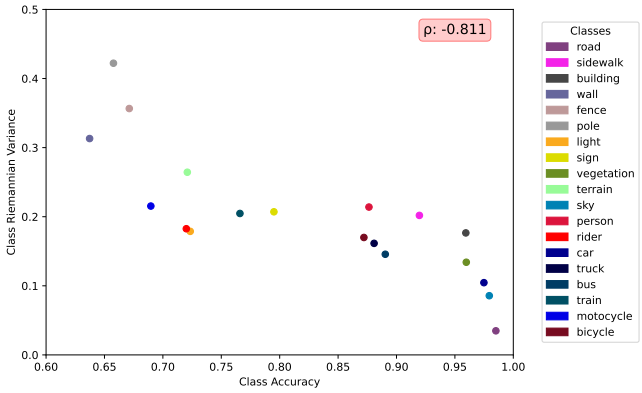

Figure 6. Plot of per-class accuracy against per-class Riemannian variance.

A.2. Additional hyperbolic formulas

Here we report established hyperbolic formulas and definitions which have used in the paper, but not shown due to space constraints.

Poincaré Distance Given two hyperbolic vectors x, y $\in \mathbb{D}_c^N$, the *Poincaré distance* represents the distance between them in the Poincaré ball and is defined as:

$$d_{Poin}(x, y) = \frac{2}{\sqrt{c}} \, tanh^{-1}(\sqrt{c}\| - x \oplus_c y\|) \tag{A1}$$

where $\oplus_c$ is the Möbius addition defined in Eq. 2 of the paper and c is the manifold curvature.

Riemannian Variance Given a set of hyperbolic vectors $x_1, ..., x_M \in \mathbb{D}_c^N$ we define the Riemannian variance between them as:

$$\sigma^2 = \frac{1}{M} \sum_{i=1}^{M} d_{Poin}^2(x_i, \mu) \tag{A2}$$

where μ is the Fréchet mean, the hyperbolic vector that minimizes the Riemannian variance. μ cannot be computed in closed form, but it may be approximated with a recursive algorithm (Lou et al., 2021).

Hyperbolic Multinomial Logistic Regression (MLR) Following Ganea et al. (2018), to classify an image feature $z_i \in \mathbb{R}^N$ we project it onto the Poincaré ball $h_i = \exp_x^c(z_i) \in \mathbb{D}_c^N$ and classify with a number of hyperplanes H_y^c (known as "gyroplanes") for each class y:

$$H_y^c = \{h_i \in \mathbb{D}_c^N, \langle -p_y \oplus_c h_i, w_y \rangle\}, \tag{A3}$$

where, p_y represents the gyroplane offset, and w_y represents the orientation for class y. The distance between a Poincaré ball embedding h_i and the gyroplane H_y^c is given by:

$$d(h_i, H_y^c) = \frac{1}{\sqrt{c}} sinh^{-1}\left(\frac{2\sqrt{c}\langle -p_y \oplus_c h_i, w_y \rangle}{(1 - c\| - p_y \oplus_c h_i\|^2)\|w_y\|}\right), \tag{A4}$$

Based on this distance, we define the likelihood as $p(\hat{y}_i = y|h_i) \propto exp(\zeta_y(h_i))$ where $\zeta_y(h_i) = \lambda_{p_y}^c \|w_y\| d(h_i, H_y^c)$ is the logit for the y class.

A.3. Implementation details

For all experiments, the model is trained on 4 Tesla V100 GPUs using PyTorch (Paszke et al., 2019) and PyTorch Lightning with an effective batch-size of 8 samples (2 per GPU). The DeepLab-v3+ architecture is initialized with an Imagenet pre-trained ResNet-101 as the backbone. *RiemannianSGD* optimizer with momentum of 0.9 and weight decay of 5×10^{-4} is used for all the trainings. The base learning rates for the encoder and decode head are 1×10^{-3} and 1×10^{-2} respectively, and they are decayed with a "polynomial" schedule with power 0.5. The models are pre-trained for 15K iterations and adapted for an additional 15K on the target set. As per (Xie et al., 2022a), the source images are resized to 1280×720, while the target images are resized to 1280×640.

A.4. Comparison of parameters count

To provide additional insights into the hyperbolic architecture employed, we conduct a comparison of parameter counts between RIPU (Xie et al., 2022a) and our method HALO (see Table A5). Both employ the DeepLab-v3+ architecture but with some distinctions. RIPU operates with a pixel embedding dimension of 512, resulting in a parameter count of 60.1M. In contrast, HALO operates with a reduced pixel embedding dimension of 64, which the adoption of a hyperbolic learning enables. Moreover, the HyperMLR requires fewer parameters than the Euclidean Linear layer used for classification due to the reduced embedding dimension. This results in a slightly lower total parameter count than RIPU's (10k fewer params). Additionally, HALO introduces the HFR module, consisting of two linear layers separated by a BatchNorm layer and a ReLU. Thanks to the lower embedding dimensions, the input and output sizes of the HFR module are only 64-dimensional, adding less than 10k additional parameters. This roughly matches the number of parameters removed from the segmenter. These modifications result in the parameter count being nearly identical between the two methods (60.1M), aligning with other studies leveraging the DeepLab-v3+ architecture.

Table A5. Comparison of parameters count in HALO vs. RIPU (Xie et al., 2022a).

Method	Segmenter	Dim.	HFR (params)	Total Params
RIPU	DeepLab-v3+	512	Not used	60.1M
HALO (ours)	Hyper-DeepLab-v3+	64	10k	60.1M

A.5. Computational Resources

We evaluated the computational resources required by our method, HALO, compared to the previous state-of-the-art, RIPU, using the setup described in Appendix A.3. The comparison was conducted under the source+target protocol.

A.5.1. Environment and Computational Load Metrics

Using an identical environment — the same conda environment, 4 Tesla V100 GPUs, and a batch size of 2 per GPU — we compared different computational load metrics for HALO and RIPU using DeepLab-v3+. Table A6 summarizes the comparison:

Table A6. Comparison of computational load metrics for HALO and RIPU (Xie et al., 2022a).

Method	FLOPS ↓	FPS ↑	Params ↓
RIPU	125.49 M	5.62	60.1 M
HALO	280.17 M	4.72	60.1 M

FLOPs (Floating Point Operations) are calculated only for the classification layers, which differ between the RIPU and HALO segmentation models. The rest of the model architectures require 1.7 TFLOPS. FPS (Frames Per Second) is measured at inference time. The parameter count (Params) refers to the entire model.

Both models have identical parameter counts and are trained using the same batch size, resulting in negligible differences in memory consumption.

A.5.2. Training and inference times

Training RIPU takes 12.5 hours, while training HALO takes 13.5 hours, which is just an 8% increase. This marginal difference is primarily due to the nature of operations in hyperbolic space, such as Möbius addition. Despite this, the increased computational cost of hyperbolic operations is offset by the reduced embedding size needed to achieve state-of-the-art performance in hyperbolic space, as detailed in Appendix A.4.

At inference time, evaluated on the Cityscapes validation set, RIPU takes 1m:29s, while HALO takes 1m:46s. Again, the difference in computing time is minor.

A.5.3. Optimization considerations

It is important to note that existing implementations of hyperbolic operations are still under active development and may not yet be fully optimized for mainstream deep learning frameworks and hardware. Specifically, the primary hyperbolic operations — exponential mapping (Eq. 1), Möbius addition (Eq. 2), and Poincaré distance (Eq. A1) — have not been optimized to the same extent as their Euclidean counterparts, which have benefited from over a decade of optimization.

This analysis demonstrates that the computational overhead introduced by hyperbolic operations is manageable and does not significantly impact the overall efficiency of our method.

A.6. Qualitative results

In Fig. 9, we present visualizations of HALO's predicted segmentation maps and the selected pixels. In the first row, HALO prioritizes the selection of pixels that are not easily interpretable, as evident in the *fence* or *wall* on the right side of the image. Notably, HALO does not limit itself to selecting contours exclusively; it continues to acquire pixels within classes if they exhibit high acquisition score. This behavior is also observed in rows 2, 3, and 4 of Fig. 9. For classes with lower complexity, such as *road* and *car*, HALO acquires only the contours. However, for more intricate classes like *pole* and *signs*, it also selects pixels within the class.

In rows 5, 6, and 8, the images depict a crowded scene with numerous small objects from various classes. Remarkably, the selection process directly targets the more complex classes (such as *pole* and *signs*), providing an accurate classification of these. In row 7, we observe an example where the most common classes (*road*, *vegetation*, *building*, *sky*) dominate the majority of the image. HALO efficiently allocates the labeling budget by focusing on the more complex classes, rather than expending resources on these prevalent ones. Refer to Sec. A.7 and Fig. 8 for a detailed overview of the selection prioritization during each active learning round.

A.7. Data acquisition strategy: rounds of selection

In this section, we analyze how the model prioritizes the selection of the pixels during the different rounds. In Fig. 7, we consider the ratio between the selected pixel at each round and the total number of pixels for the considered class. Note how the model selects in the early stages from the class with high intrinsic difficulty (e.g., rider, bicycle, pole). During the different rounds, the number of selected pixels decreases because of the scarcity of pixels associated with these classes. On the other hand, less complex classes are less considered in the early stages and the model selects from them in the intermediate rounds if the class has an intermediate complexity (e.g., wall, fence, sidewalk) or in the last stages if it has low complexity (e.g., road or building).

The qualitative samples of pixel selections in Fig. 8 corroborate this observation. In rounds 1 and 2, the model gives precedence to selecting pixels from more complex classes (e.g., *poles*, *sign*, *person*, or *rider*). Subsequently, HALO shifts its focus to two distinct objectives: i) acquiring contours from classes with lower complexity (e.g., *road*, *car*, or *vegetation*), and ii) obtaining additional pixels from more complex classes (e.g., *pole* or *wall*). Notably, in rows 1, 2, 3, 5, and 6, HALO gives priority to selecting complete objects right from the initial round (as seen with the *sign*). Another noteworthy instance is the acquisition of the *bicycle* in row 7. The hyperbolic radius score enables the acquisition of contours that extend beyond the boundaries of pseudo-label classes. In this case, we observe precise delineation of the internal portions of the wheels.

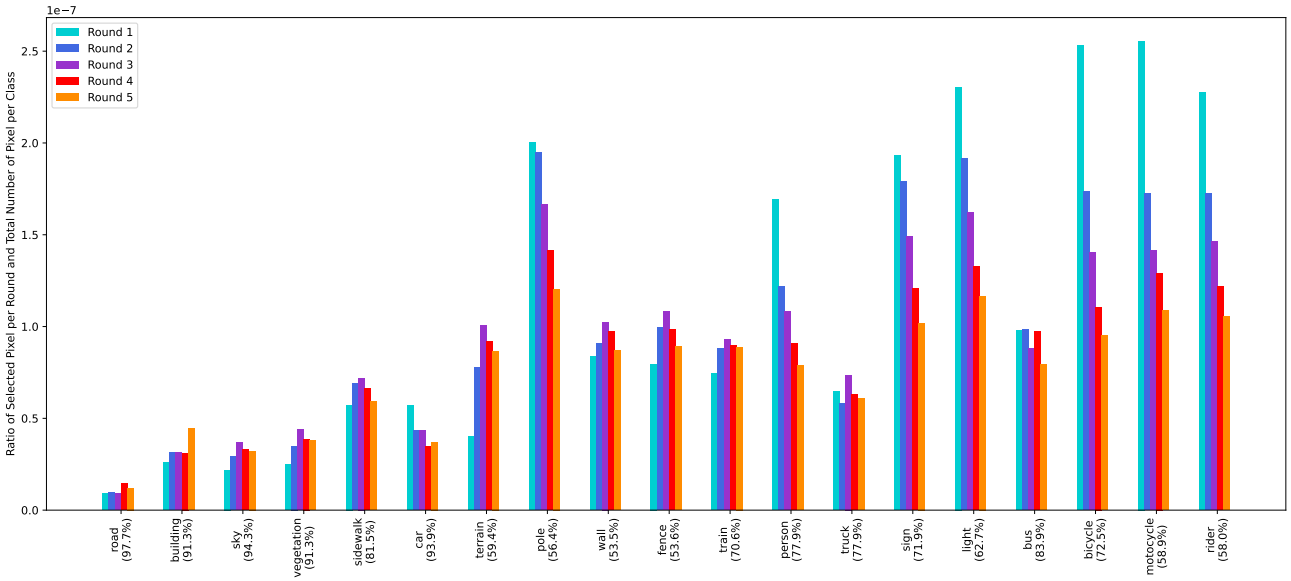

Figure 7. Ratio between the selected pixels for each class at each round and the total number of pixels per class. Each color shows the ratio in the specific round. On the x-axis are reported the classes with the relative mIoU (%) of HALO (cf. Table 1 of the main paper) ordered according to they decreasing hyperbolic radius.

A.8. Qualitative comparison with the baseline model

The top row of Fig. 10 depicts label acquisition using the baseline RIPU method with budgets of 2.2% (left) and 5% (right). The bottom row illustrates visualizations with our proposed HALO using the same budgets. Noteworthy observations include:

- By design, RIPU only concentrates on selecting boundaries between semantic parts (ref. Fig. 10 top-left). However, since there are only a few (thin) boundary pixels, RIPU soon exhausts the pixel selection request. Next, when a larger budget is available, RIPU simply samples from the left side. The random selection still provides additional labels (ref. Fig. 10 top-right) and is a good baseline, cf. Table 3 of Xie et al. (2022a), although not as good as HALO's acquisition strategy.
- By contrast, HALO showcases pixel selection from both boundaries and internal regions within semantic parts (ref. Fig. 10 bottom-left). Especially passing from 2.2% to 5% acquisition budget, HALO considers thick boundaries, so also parts of objects close to the boundaries, but also areas within objects, as it happens for wall, fence, pole, and sidewalk, cf. the right image part in the bottom-right of Fig. 10.

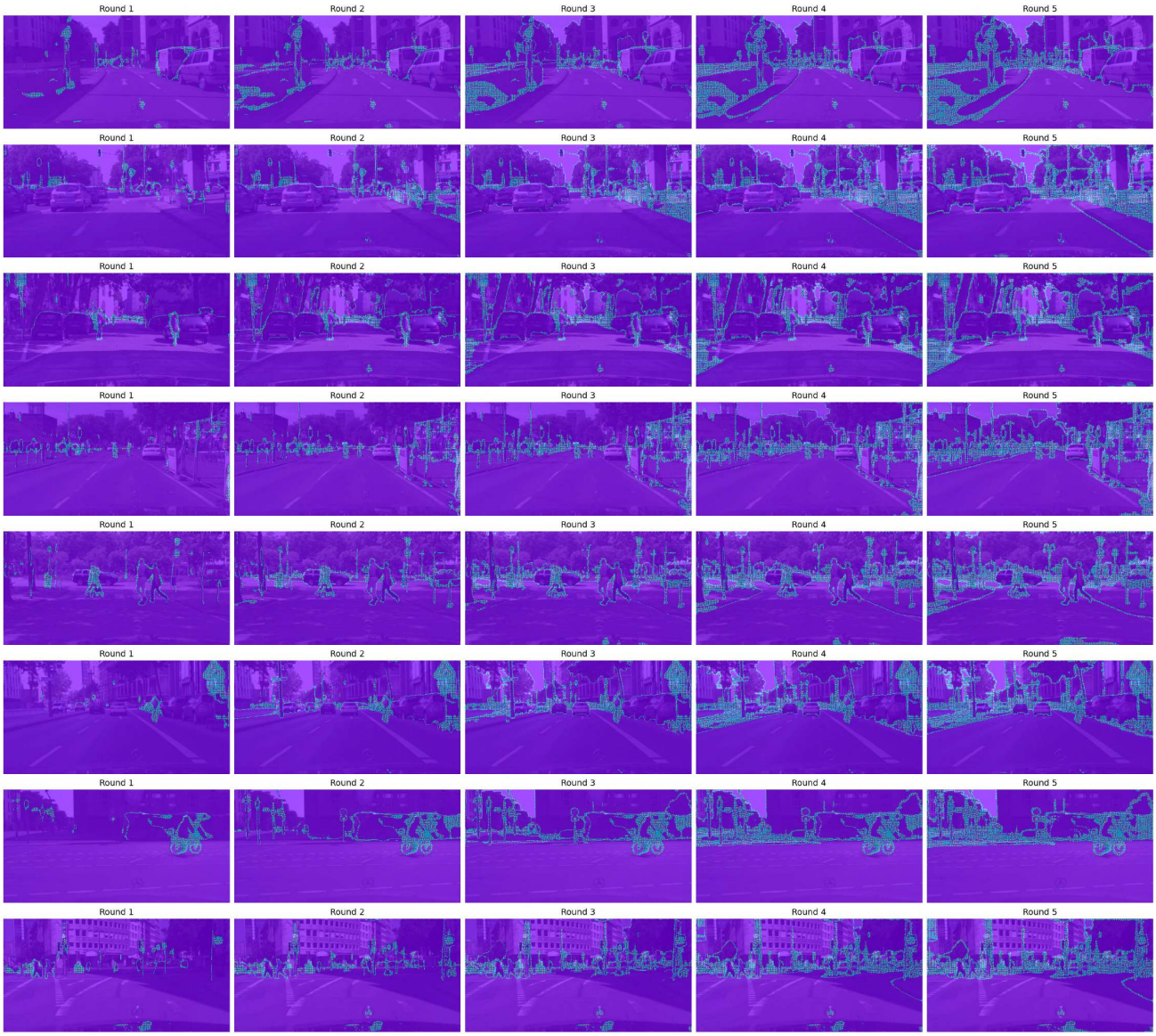

Figure 8. Qualitative analysis on the pixel selected by HALO at each round. Zoom in to see details.

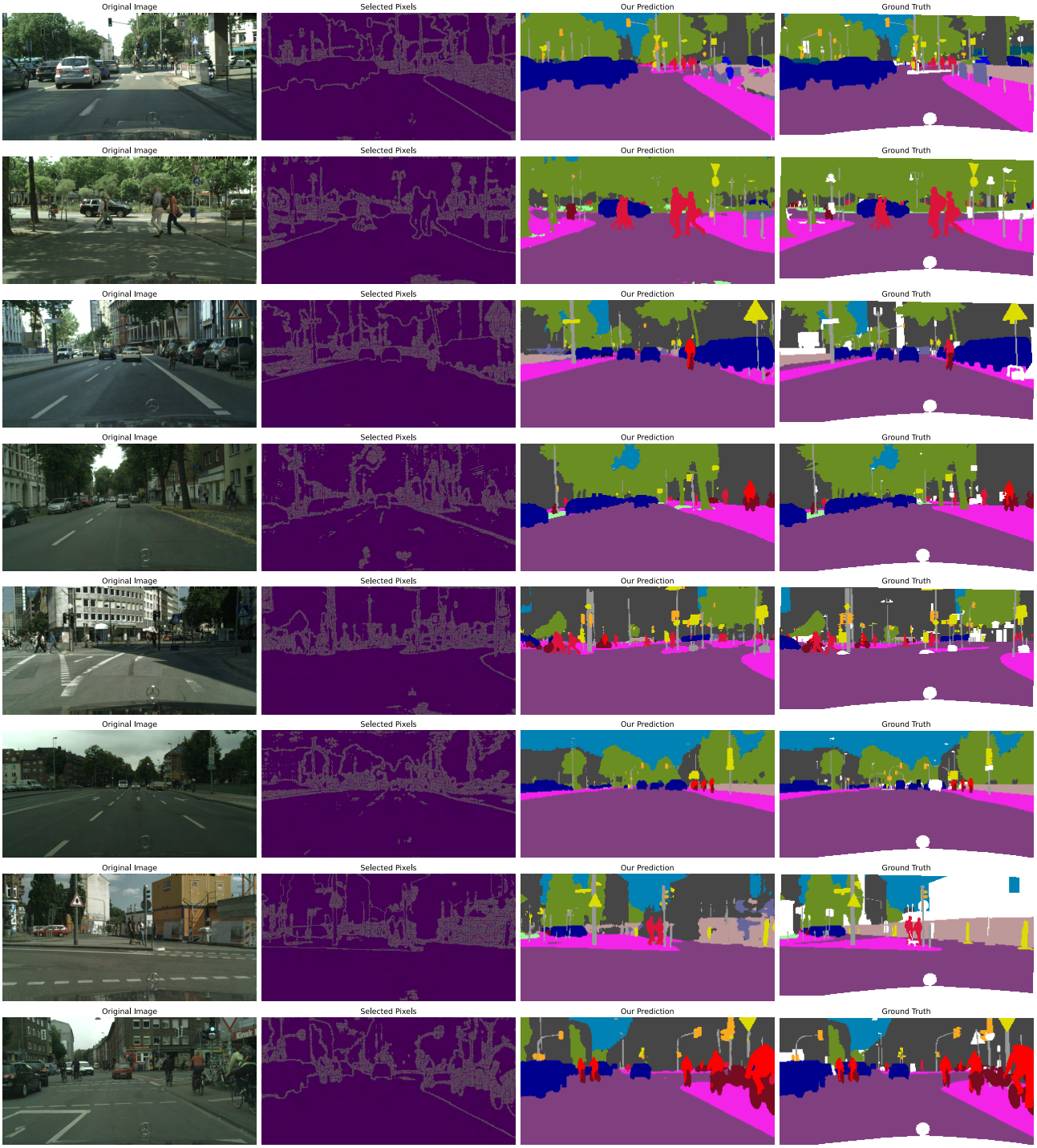

Figure 9. **Qualitative Results Visualization for the GTAV → Cityscapes Task.** The figure showcases different subfigures representing: the original image, HALO's pixel selection, HALO's prediction, and the ground-truth label. Zoom in for the details.

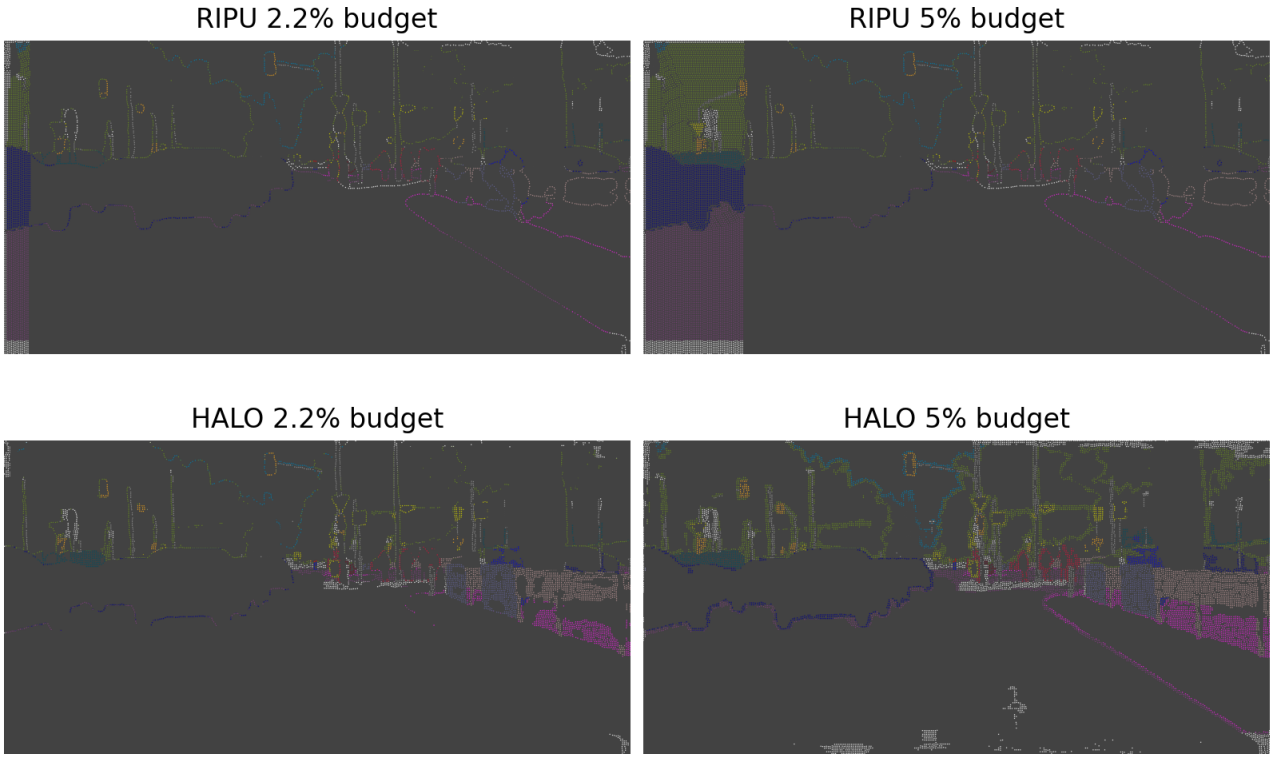

Figure 10. (top-row) Pixel selection with RIPU's baseline; (bottom-row) Pixel selection with out HALO; (left-column) Selection with budget 2.2%; (right-column) Selection with budget 5%. Zoom in for the details.