Learning Affinity with Hyperbolic Representation for Spatial Propagation

Jin-Hwi Park [1] **Jaesung Choe** [2] **Inhwan Bae** [1] **Hae-Gon Jeon** [1]

Abstract

Recent approaches to representation learning have successfully demonstrated the benefits in hyperbolic space, driven by an excellent ability to make hierarchical relationships. In this work, we demonstrate that the properties of hyperbolic geometry serve as a valuable alternative to learning hierarchical affinity for spatial propagation tasks. We propose a Hyperbolic Affinity learning Module (HAM) to learn spatial affinity by considering geodesic distance on the hyperbolic space. By simply incorporating our HAM into conventional spatial propagation tasks, we validate its effectiveness, capturing the pixel hierarchy of affinity maps in hyperbolic space. The proposed methodology can lead to performance improvements in explicit propagation processes such as depth completion and semantic segmentation.

1. Introduction

The goal of an affinity map is to model the pixel-wise relations of given input images for low-level vision tasks, such as the image segmentation task (Shi & Malik, 2000; Jiang et al., 2018; Liu et al., 2017) and the scene depth computation (Cheng et al., 2018; Park et al., 2020; Cheng et al., 2020; Lin et al., 2022; Choe et al., 2021; Shin et al., 2023), etc. In early works (Weickert, 1998; Levin et al., 2004; Yatziv & Sapiro, 2006; Farbman et al., 2010; Shi & Malik, 2000), the pixel-wise relation is based on parametric models driven by low-level information in images, including distinctive local keypoints or edge boundaries. With the advent of convolution neural networks (CNNs), it is now feasible to define pixel-wise affinity, considering scene contexts using learned parameters and convolution kernels.

Recently, learning affinity from CNNs was primarily developed for spatial propagation tasks, called Spatial Propagation Networks (SPNs) (Maire et al., 2016; Liu et al., 2017; Bertasius et al., 2017). Starting with these pioneering studies, several works (Cheng et al., 2018; Jiang et al., 2018) have addressed an inherent problem of existing SPNs, limited local receptive fields. To solve this problem, the studies adopt either pyramidal encoder-decoder networks (Jiang et al., 2018) or iterative propagation schemes (Cheng et al., 2018).

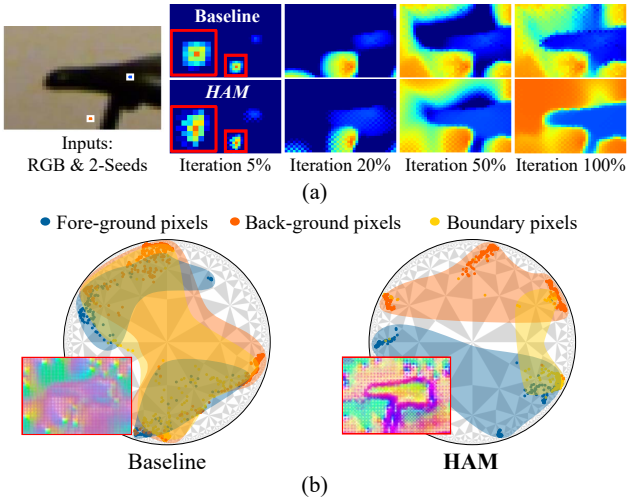


Figure 1. Comparison results between baseline and HAM. (a) Examples of spatial propagation from a conventional method (Cheng et al., 2018) and our method with two initial seeds and an RGB image. (b) Visualizing pixel features to construct affinity map on the 2D Poincaré ball using UMAP (McInnes et al., 2018). The red boxes indicate affinity visualization using PCA (Jolliffe, 1986). Further details and examples are reported in Sec. 5.3.

Nonetheless, as shown in Fig. 1-(a), these works still suffer from the bleeding error at object boundaries where noises or smooth intensity changes occur in images. This demonstrates that ill-defined affinities cause inevitable errors by invading another region, especially the nearby boundary.

To alleviate the vagueness of measuring pixel affinity, a hierarchical structure was proposed as a solution in several pioneer studies. Specifically, previous studies build tree structures for the pixel distance computation to design an edge-preserving filter (Bao et al., 2013; Dai et al., 2015), conduct non-local cost aggregation (Yang, 2012; 2014), and measure the image boundary connectivity (Tu et al., 2016). Unfortunately, these interesting ideas cannot be formulated

[1]AI Graduate School, GIST, South Korea [2]KAIST, South Korea. Correspondence to: Hae-Gon Jeon <haegonj@gist.ac.kr>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

to address the problem in conventional SPNs because it is hard to embed tree structures into Euclidean space without high distortion, as demonstrated in (Linial et al., 1995).

In terms of hierarchical representation, hyperbolic embedding is well-known to continuously embed tree structures with arbitrarily low distortion (Krioukov et al., 2010; Sarkar, 2012) and has been proposed as an alternative way to learn hierarchical representations and graph-structured data (Nickel & Kiela, 2017; 2018; Tifrea et al., 2019). Compared to Euclidean space, the volume of hyperbolic space grows exponentially with the radius, allowing exponentially-growing hierarchies and tree-like structures to be embedded with low distortion (Sala et al., 2018). By adopting hyperbolic embedding in learning frameworks, hierarchical relations can be further formulated due to their ability to represent hierarchical data.

Taking full advantage of the hyperbolic property, in this paper, we first propose a Hyperbolic Affinity learning Module (HAM) to construct *hierarchical pixel affinity* maps for spatial propagation tasks. Our design relies on two key insights: (1) applying the hyperbolic geometry into the image pixel domain (2) specializing the hyperbolic neural operation to account for the hierarchical relationship among pixels. In particular, while a conventional operation in Euclidean space cannot impose a hierarchical relation on pixels, HAM can construct a priority-based hierarchy using our novel geodesic weight and β-priority. The geodesic weight provides more attention to the geodesic-closest pixels to keep the hierarchical structure and selectively aggregate important pixels. The β-priority rearranges the pixels along with the hyperbolic distances, which implicitly supports the positional information for transformed pixel features. To demonstrate the validity and efficacy, we conduct extensive experiments and analysis on spatial propagation tasks, including depth completion and semantic segmentation, even with the same number of parameters as traditional SPNs.

2. Related Work

Spatial Propagation Network. Propagating initial seeds (*i.e.,* user-scribble or initial prediction) and finding the optimal groupings of pixels (*i.e.,* affinity maps) are necessary for low-level vision tasks: image segmentation (Shi & Malik, 2000), object semantic segmentation (Jiang et al., 2018; Liu et al., 2017), colorization (Levin et al., 2004), video recognition (Wang et al., 2018), and depth completion (Cheng et al., 2018; 2019). With the advent of CNN era, learning affinities from CNNs for spatial propagation has received high interest. Due to the hierarchical nature of the features (Zeiler & Fergus, 2014), SPNs learn task-specific affinity with high-level features inferred with feature extractor (Fig. 4). This mechanism is successfully applied to various vision tasks with the highly engineered implementation of CNNs, such as depth completion (Cheng et al., 2018; Park et al., 2020; Lin et al., 2022) and semantic segmentation (Liu et al., 2017; Jiang et al., 2018; Bertasius et al., 2017). Despite their success, inferred affinity maps have difficulty handling boundary ambiguities due to an inherent problem of the conventional convolution operations that only cover grid data in the Euclidean space.

Hyperbolic Neural Network. Existing neural architectures that utilize hyperbolic geometry can be divided into two approaches. The first approach focuses on learning hyperbolic embeddings that lead to promising performances in various Natural Language Processing (NLP) (Nickel & Kiela, 2018; Tifrea et al., 2019; Nickel & Kiela, 2017). The second approach establishes deep hyperbolic neural networks, whose representative works include the hyperbolic multi-layer perceptrons (Ganea et al., 2018), hyperbolic graph convolutional neural networks (Dai et al., 2021; Chami et al., 2019; Liu et al., 2019), hyperbolic attention networks (Gulcehre et al., 2019), and hyperbolic convolution layers (Ryohei et al., 2021).

In the visual perception tasks, several studies have shown that the hyperbolic embeddings can provide a better alternative (*e.g.,* image few-shot (Khrulkov et al., 2020; Gao et al., 2021; Ma et al., 2022), action search (Long et al., 2020) metric learning (Yan et al., 2021; Ermolov et al., 2022), 3D voxel-grid biomedical image (Hsu et al., 2021), and semantic segmentation (GhadimiAtigh et al., 2022)). The most relevant work to this paper would be (GhadimiAtigh et al., 2022) which takes hyperbolic representation to the pixel-level; however, they concentrate on hierarchical relations among semantic classes, *i.e.,* label hierarchy as (Liu et al., 2020; Long et al., 2020). By contrast, we design a new convolutional operation in the hyperbolic space to represent hierarchical property at the pixel-level, and it can be plugged into the standard SPNs (Jiang et al., 2018; Cheng et al., 2018) without any additional learnable parameters.

3. Mathematical Preliminaries

In this section, we review a definition of a hyperbolic embedding on a Poincaré ball and the details of the fundamental arithmetic operations (Sec. 3.1). We then discuss the reason why hyperbolic geometry is effective for the spatial propagation task (Sec. 3.2). Lastly, we provide a quantification analysis of hyperbolicity that indicates a tree-likeliness of the embedded features to verify the validity of utilizing hyperbolic representation for pixel affinity construction (Sec. 3.3).

3.1. Background of Hyperbolic Geometry

The hyperbolic space is a Riemannian manifold with a constant negative sectional curvature equipped with hyperbolic geometry. To model this space, we follow a Poincaré ball

model to differentiably connect Euclidean space and hyperbolic space, which is employed in most preceding works (Nickel & Kiela, 2017; Ryohei et al., 2021; Khrulkov et al., 2020; Ermolov et al., 2022; GhadimiAtigh et al., 2022). The Poincaré ball model $(\mathbb{D}_\kappa^n, \mathfrak{g}^\kappa)$ with curvature κ is defined by a manifold $\mathbb{D}_\kappa^n = \{x \in \mathbb{R}^n \mid \kappa\|x\| < 1\}$ equipped with a metric $\mathfrak{g}^\kappa$, where $\|\cdot\|$ denotes the Euclidean norm. The induced distance between two points $u, v \in \mathbb{D}_\kappa^n$ is given by

$$d_\kappa(u,v) = \frac{1}{\sqrt{\kappa}} \cosh^{-1}\left(1 + \frac{2\kappa\|u-v\|^2}{(1-\kappa\|u\|^2)(1-\kappa\|v\|^2)}\right), \tag{1}$$

Since hyperbolic spaces are not vector spaces in a traditional sense, we use the formalism of Möbius gyrovector space (Ungar, 2008; 2001) which is a generalization of Euclidean vector spaces to models of hyperbolic space.

Möbius addition. For a pair $(u, v) \in \mathbb{D}_\kappa^n$, the equation of the Möbius addition is defined as follows:

$$u \oplus_\kappa v = \frac{(1 + 2\kappa\langle u, v\rangle + \kappa\|v\|^2)u + (1 - \kappa\|u\|^2)v}{1 + 2\kappa\langle u, v\rangle + \kappa^2\|u\|^2\|v\|^2}, \tag{2}$$

where $\langle\cdot,\cdot\rangle$ is the Euclidean inner product.

Exponential and logarithmic mapping. To emembed Euclidean vectors into the hyperbolic space, one first needs to define a mapping function from $\mathbb{R}^n$ to $\mathbb{D}_\kappa^n$, and vice versa. The exponential and the logarithmic mapping are bijective functions which have appealing forms at an origin, namely for $\mathbf{x} \in \mathbb{R}^n$ and $u \in \mathbb{D}_\kappa^n$:

$$\exp_0^\kappa(\mathbf{x}) = \tanh(\sqrt{\kappa}\|\mathbf{x}\|)\frac{\mathbf{x}}{\sqrt{\kappa}\|\mathbf{x}\|}, \tag{3}$$

$$\log_0^\kappa(u) = \tanh^{-1}(\sqrt{\kappa}\|u\|)\frac{u}{\sqrt{\kappa}\|u\|}, \tag{4}$$

Möbius multiplication. For an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}^m$ in Euclidean space, the Möbius version of f is a function that maps from $\mathbb{D}^n$ to $\mathbb{D}^m$ in the hyperbolic space using Eq (3). Similarly, we can derive the Möbius matrix-vector multiplication between the matrix M and input $\mathbf{u}$, which is defined as:

$$M \otimes_\kappa u = (1/\sqrt{\kappa})\tanh\left(\frac{\|Mu\|}{\|u\|}\tanh^{-1}(\sqrt{\kappa}\|u\|)\right)\frac{Mu}{\|Mu\|}. \tag{5}$$

Note that the Möbius scalar multiplication also can be obtained by projecting x in the tangent space at 0, multiplying this projection by the scalar in the tangent space.

3.2. Rationale: Hyperbolic Representation for Affinity

In order to explain the hyperbolic representation as being a pixel affinity module, we offer theoretical insights, specifically highlighting the essential role of hyperbolic geometry in pixel-wise relationships through remarks and conjecture.

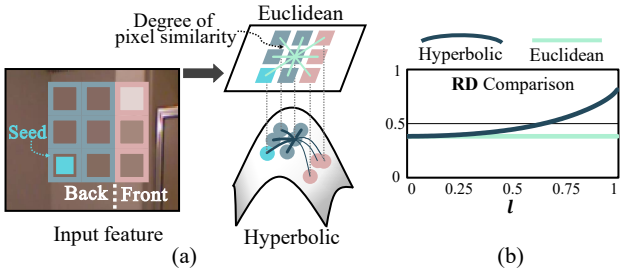


Figure 2. (a) An illustration of the feature embedding into Euclidean and hyperbolic space. (b) Comparison of the ratio distance (RD, Eq (8)) between Euclidean and hyperbolic space. l denotes the distance from the origin, $||p|| = ||q|| = l$.

Remark 3.1. **[Distinction property]** Let n-dimensional $x, y \in \mathbb{R}^n$ are unit vectors and points u, v are obtained by projecting x, y from the tangent space at 0 with the exponential map Eq (3) in Poincaré ball $\mathbb{D}_1^n$ with curvature 1. Since the x, y are unit vectors, the points $u, v \in \mathbb{D}_1^n$ are defined as below:

$$u = tx, \quad v = ty \tag{6}$$

where $t = tanh(1) \simeq 0.7616$. Now, we can compute the hyperbolic distance $d_{\kappa=1}$ with Euclidean distance $d_E = ||x - y||$ as below:

$$d_{\kappa=1}(u,v) \simeq cosh^{-1}(1 + 6.5774 * d_E(x,y)^2) \tag{7}$$

Since x, y are unit vectors in Euclidean space, the distance ranges from 0 to 2. According to the Eq (7), if d_E is close to 0, which means that two pixels belong to the same label in the spatial propagation, $d_{\kappa=1}$ also has almost zero values. In contrast, if the two pixels are irrelevant, the hyperbolic distance has at least 2x larger distances than the Euclidean distance between corresponding points. In practice, the bleeding error occurs when gradients at edge boundaries are smoothly changed. Here, hyperbolic feature embedding is helpful for alleviating vagueness by boosting the distinction. Note that this **distinction property** of hyperbolic features can be observed in Fig. 1-(b) and Fig. 2-(a).

Remark 3.2. **[Exponential growth property]** The surface area of an $(n-1)$-dimensional sphere of radius r in n-dimensional Euclidean space $\mathbb{R}^n$ is $S_{\mathbb{R}^n}(r) = \alpha_n r^{n-1}$, and n-dimensional hyperbolic space $\mathbb{H}^n$ is $S_{\mathbb{H}^n}(r) = \alpha_n \sinh(r)^{n-1}$ where $\alpha_n = \pi^{n/2}/(n/2)!$ is the volume of a ball of unit radius. The growth of the surface is polynomial in Euclidean space, but it is **exponential** ($S_{\mathbb{H}^n}(r) \sim \alpha_n e^{r(n-1)}$) for hyperbolic space.

To ease understanding of the exponentially growing hyperbolic distance, we can consider two points, p and q, on a unit disk, which have the same length from the origin ($||p|| = ||q|| = l$) and its ratio distance (RD) is defined as:

$$RD(p,q) = \frac{d_{E/H}(p,q)}{d_{E/H}(q,0) + d_{E/H}(p,0)}, \tag{8}$$

Table 1. Calculate relative δ-hyperbolicity on various dataset using learned features to construct affinity or similarity map within SPNs. Results are averaged across testset of size 1000 and the standard deviation for all the experiments did not exceed 0.002.

	Depth Completion			Semantic Segmentation		
	NYUv2	KITTI	ScanNet	Pascal-VOC	Pascal-Context	ADE20K
δ_{hyp}	0.326	0.263	0.271	0.237	0.198	0.259

where d_E and d_H denote Euclidean and hyperbolic distance (Eq (1)), respectively. As these points move toward the outside of the disk, Euclidean and hyperbolic spaces have different RD values. According to Fig. 2-(b), the ratio value is constant in Euclidean space; however, it goes to 1 in the hyperbolic space. This nature of hyperbolic space is the key property to constructing robust affinity in the hyperbolic space by enlarging the distance (*i.e.,* low affinity).

Conjecture 3.3. *Hyperbolic representation alleviates the bleeding problem by enhancing the distinction between unrelated pixel features. The* ***distinction property*** *(Remark 3.1) and the* ***exponential growth property*** *(Remark 3.2) comparing Euclidean space guarantee high-fidelity pixel relations for spatial propagation tasks.*

3.3. Pixel-level Hyperbolicity

To compute the hyperbolicity of the pixel affinity map, which is used to measure a "*tree-likeness*", we follow the quantitative analysis described in (Khrulkov et al., 2020; Ermolov et al., 2022) to show the efficacy of the hyperbolic embedding of the feature map extracted from images. They adopt a relative δ hyperbolicity, $\delta_{hyp}(X)$, of which low value denotes that the set X has an underlying hyperbolic geometry, *i.e.,* it is an approximately tree-like structure. As shown in Tab. 1, the δ_{hyp} are significantly close enough to 0. Therefore, it seems to be an appropriate choice to apply hyperbolic embedding into pixel affinity construction. Further details are reported in the supplemental materials.

4. Hyperbolic Affinity Learning

In this work, we design a hyperbolic neural operation to take advantage of the representation power of the hyperbolic geometry. By embedding pixel features to hyperbolic space and aggregating them, we can embed hierarchical property in the pixel affinity. Instead of adopting a graph neural network such as prior works (Chami et al., 2019; Liu et al., 2019), we design a hyperbolic convolution operation to utilize a local inductive bias and follow the affinity construction mechanism of previous SPNs, which adopt simple 2D convolution layer to learn task-specific affinity.

For this, we first define the naïve hyperbolic convolution (Ryohei et al., 2021) using concatenation operation and a fully connected layer in the hyperbolic space (Sec. 4.1). To enhance the hierarchical relationships among pixel features, we propose a specialized hyperbolic convolutional operation for robust spatial propagation, called HAM (Sec. 4.2). We lastly incorporate our method into the conventional SPN formulations (Sec. 4.3). The overall algorithm scheme is described in Fig. 3 and Algorithm 1.

Algorithm 1 Hyperbolic Affinity Learning Module

Input: image features $\mathcal{F}$, a set of signed distance from a center of the conv weight Ω, a bias term $\mathbf{b}$, and convolution kernel matrix $\mathbf{W}$.

Function HAM($\mathcal{F}$, Ω, $\mathbf{b}$, $\mathbf{W}$)

for $\mathbf{f}_{(x,y)}$ in $\mathcal{F}$ **do**
  Projection: $\mathbf{h}_{(x,y)} = \mathcal{M}(\mathbf{f}_{(x,y)})$
  Concatenation: $[h_1, ...] = \mathcal{C}^{\text{geo}}_{(i,j)\in\Omega}(\mathbf{h}_{(x+i,y+j)})$
  Geo-weight: $[\bar{g}_1, ...] = \Delta d_\kappa(\mathbf{h}_{(x,y)}, [\bar{h}_{(i,j)}])$
  Convolution: $\tilde{\mathbf{h}}_{(x,y)} = \mathbf{W} \otimes_\kappa [\bar{g}_1 \otimes_\kappa \bar{h}_1, ...] \oplus_\kappa \mathbf{b}$
end for

output Hyperbolic affinity features $\tilde{\mathbf{h}}_{(x,y)} \in \widetilde{\mathbf{H}}$

4.1. Hyperbolic Convolution Layer

Given image feature maps $\mathcal{F}$ in Euclidean space, we pixel-wisely embed an image feature vector at a pixel (x, y) (*i.e.,* $\mathbf{f}_{(x,y)} \in \mathbb{R}^C$) into the hyperbolic space. Here, we utilize an exponential mapping $\mathcal{M}(\cdot) = \exp_0^\kappa(\cdot)$ on the Poincaré ball $\mathbb{D}^C_\kappa$ as a bijective function between the Euclidean space and the hyperbolic space via Poincaré curvature κ.

The most intuitive way for the hyperbolic neural operation to construct a pixel affinity map is to apply a conventional convolution into features after passing through the bijective mapping functions. However, to fully take advantage of the hyperbolic representation, we concatenate hyperbolic features to regularize expected values of vector norms and aggregate them using Poincaré fully-connected layer. To operate the hyperbolic convolution in these manners, we adopt a generalization technique proposed in (Ryohei et al., 2021) with a coefficient $\beta_n = B(\frac{n}{2}, \frac{1}{n})$, where B is a Beta distribution as below:

$$\mathcal{C}^\beta(\mathbf{h}_1, \dots \mathbf{h}_N) = \mathcal{M}\Big((\beta_n \beta_{n_1}^{-1} \mathbf{f}_1^T,\ ...,\ \beta_n \beta_{n_N}^{-1} \mathbf{f}_N^T)^T\Big), \tag{9}$$

The points $\mathbf{h}_i$ in the Poincare ball $D_\kappa^{n_i}$ are projected back $\mathbf{f}_i = \mathcal{M}^{-1}(\mathbf{h}_i)$ with the scalar coefficient β_n. Note that N indicates the number of concatenated points, $n = \sum_{i=1}^N n_i$.

Then, we can formulate a naïve hyperbolic convolution with hyperbolic feature $\mathbf{h}_{(x,y)} = \mathcal{M}(\mathbf{f}_{(x,y)})$ for a 2-dimensional image domain as below:

$$\hat{\mathbf{h}}_{(x,y)} = \mathbf{W} \otimes_\kappa \mathcal{C}^\beta_{(i,j)\in\Omega}(\mathbf{h}_{(x+i,y+j)}) \oplus_\kappa \mathbf{b}, \tag{10}$$

where $\mathbf{W} \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}} \times \gamma \times \gamma}$ is a convolution weight ma-

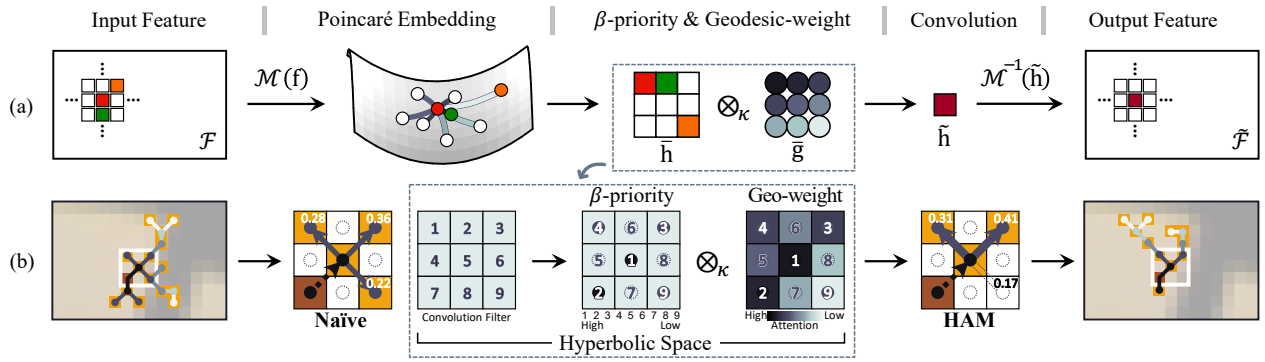


Figure 3. (a) Convolutional neural operation on the hyperbolic space utilizing Poincaré ball model. For instance, the green feature is the nearest neighbor to the center feature (red) within a hyperbolic convolutional kernel. The orange point represents the adjacent pixel in Euclidean space and the farthest neighbor in hyperbolic space. (b) Comparison propagation mechanism between hyperbolic and Euclidean approach. The β-priority enhances the positional relation in the hyperbolic operation, and the Geo-weight prioritizes semantically close pixel features. This allows HAM to achieve well-structured hierarchical affinity compared to Euclidean and naïve approaches.

trix whose kernel size is γ, and $\mathbf{b} \in \mathbb{D}_{\kappa}^{C_{\text{out}}}$ is a bias term. $\Omega{=}\{(i,j){\in}\mathbb{Z}^2 \mid (-\gamma',-\gamma'),...,(\gamma',\gamma'),\ \gamma'{=}\lfloor\frac{\gamma}{2}\rfloor\}$ is a set of signed distances from a center of the convolution kernel to others. In particular, given $\mathbf{h}_{(x,y)}$ whose channel length is C_{in} on the Poincaré ball, the features $\{\mathbf{h}_{(x',y')} \mid (x',y') \in \Omega\}$ are concatenated using the β-concatenation. Then, naïve hyperbolic convolution proceeds the convolution operation using $\mathcal{C}^{\beta}_{(i,j)\in\Omega}(\mathbf{h}_{(x',y')})$ and convolution weight $\mathbf{W}$.

4.2. Hyperbolic Affinity Learning Module

The Eq (10) assumes that the hyperbolic features $\mathbf{h}$ are uniformly distributed in the hyperbolic space as if pixel features $\mathbf{f}$ are on regular grids in the image domain. After we embed pixel features into hyperbolic space with specific curvatures, these features can actually be irregular and unordered in hyperbolic space. In other words, the inherent spatial relationship among pixels on the image domain is distorted in hyperbolic space. To alleviate the issue, we propose a β-priority and a geodesic weight, which considers the relative importance of among pixels in hyperbolic space and explicitly encourages semantically close features.

β-priority. In the Poincaré ball, hyperbolic features can have hierarchical properties (Sala et al., 2018; Nickel & Kiela, 2017), and their similarity can be measured using a normalized feature distance. It implies that a similarity between pixel features $\mathbf{f}$ can be different from the similarity between hyperbolic features $\mathbf{h}$. To further exploit their hierarchical relations in hyperbolic space, we need to assign a priority for the closer hyperbolic features $\mathbf{h}{=}\mathcal{M}(\mathbf{f})$. Accordingly, we apply a hyperbolic convolution along with their inverse distances as:

$$\left[\bar{\mathbf{h}}_1,...,\bar{\mathbf{h}}_{\gamma^2}\right] = \mathcal{C}^{\text{geo}}_{(i,j)\in\Omega}(\mathbf{h}_{(x+i,y+j)}), \tag{11}$$

where $\mathcal{C}^{\text{geo}}_{(i,j)\in\Omega}(\cdot)$ is a sorted concatenation that rearranges features in the order of the normalized distance between a reference pixel (x,y) and its neighbors $(x+i,y+j)$, where $\bar{\mathbf{h}}_i$ denotes the i-th closest neighbor (Fig. 3-(b)). The operation allows informative feature selection, endowing convolution filters in hyperbolic space with consistent positional relations.

Geodesic weight. In addition, we assign weight values based on the geodesic distances in the convolution operation. HAM learns to consider hierarchical relationships between a reference pixel and aggregated pixels through the additional weights. To do this, we normalize the geodesic distances among pixels and adopt weighted aggregation $\bar{\mathbf{g}}_{(x+i,y+j)} \otimes_{\kappa} \bar{\mathbf{h}}_{(x+i,y+j)}$ as follows:

$$\bar{\mathbf{g}}_{(x+i,y+j)} = \Delta d_{\kappa}(\mathbf{h}_{(x,y)}, \mathbf{h}_{(x+i,y+j)}), \tag{12}$$

where $\Delta d_{\kappa}(\cdot,\cdot)$ is a normalized distance between two points on the Poincaré ball defined as:

$$\frac{\exp(d_{\kappa}(\mathbf{h}_{(x,y)}, \mathbf{h}_{(x+i,y+j)}))}{\sum_{(i',j')\in\Omega}\exp(d_{\kappa}(\mathbf{h}_{(x,y)}, \mathbf{h}_{(x+i',y+j')}))}, \tag{13}$$

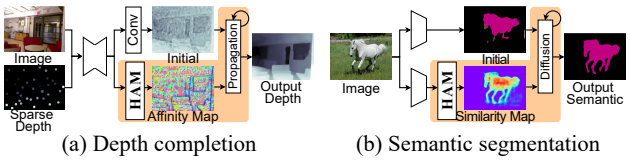
With our β-priority and geodesic weight, HAM takes advantage of the hierarchical representations $\tilde{\mathbf{h}}_{(x,y)}$ in the hyperbolic space modeled with the Poincaré ball, which is formulated as follows:

$$\tilde{\mathbf{h}}_{(x,y)} = \mathbf{W}\otimes_{\kappa}\mathcal{C}^{\text{geo}}_{(i,j)\in\Omega}(\bar{\mathbf{g}}_{(x+i,y+j)} \otimes_{\kappa} \bar{\mathbf{h}}_{(x+i,y+j)})\oplus_{\kappa}\mathbf{b}, \tag{14}$$

The aggregated features $\tilde{\mathbf{h}}_{(x,y)}$ are back-projected into the Euclidean space, $\mathcal{M}^{-1}(\tilde{\mathbf{h}}_{(x,y)})$. Compared to the naïve hyperbolic convolutional operation (Eq (10)) that follows pre-defined pixel orders in the Euclidean space, the proposed HAM assigns higher weights to closely located points in hyperbolic space. As a result, with geodesic information, our method can focus more on spatial connectivity.

Table 2. Quantitative results of depth estimation on NYUv2 (Silberman et al., 2012), Virtual-KITTIv2 (Cabon et al., 2020), and ScanNet (Dai et al., 2017). CSPN (Cheng et al., 2018) is a baseline architecture for both the naïve and HAM. (Unit: meter, **Bold: Best)**

	NYUv2						ScanNet						Virtual-KITTIv2					
	RMSE	MAE	iRMSE	iMAE	REL	$\delta^1_{1.25}$	RMSE	MAE	iRMSE	iMAE	REL	$\delta^1_{1.25}$	RMSE	MAE	iRMSE	iMAE	REL	$\delta^1_{1.25}$
CSPN	0.116	0.048	0.018	0.007	0.017	0.993	0.080	0.027	0.027	**0.009**	0.014	**0.993**	12.233	8.261	0.035	0.023	0.606	0.529
Naïve	0.108	0.043	0.017	**0.006**	0.015	**0.994**	0.078	0.027	**0.026**	**0.009**	0.014	**0.993**	10.946	7.266	0.034	0.022	0.539	0.542
Ours	**0.102**	**0.036**	**0.016**	**0.006**	**0.014**	**0.994**	**0.073**	**0.024**	0.027	**0.009**	**0.013**	**0.993**	**9.612**	**6.661**	**0.030**	**0.021**	**0.489**	**0.561**



(a) Depth completion (b) Semantic segmentation

Figure 4. The orange-colored areas indicate propagation parts where we replace the original affinity module with our HAM.

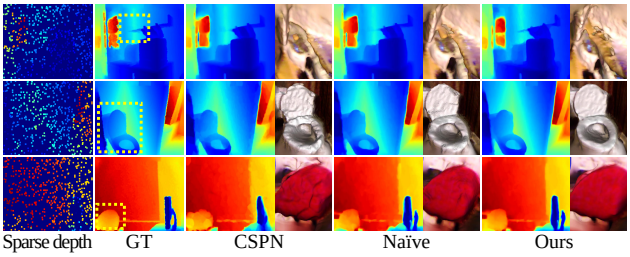


Figure 5. Qualitative comparison on NYUv2 dataset (Silberman et al., 2012) for depth completion. We display predicted depth and their 3D mesh results. It shows that HAM helps to compute correct dense depth in various shapes of regions, including slanted objects, homogeneous surfaces, and thin structures.

HAM constructs pixel-level hierarchies through the proposed hyperbolic embeddings based on their relation to the hyperbolic space as shown in Fig. 1-(b). We observe that well-separated foreground, background, and boundary pixels can be achievable from ours unlike the naïve approach.

4.3. Incorporation of HAM with SPNs

We introduce the incorporation of HAM into conventional CNN-based SPNs. For each propagation step t and an embedded pixel $\mathbf{p}$ at (x, y), we formulate a spatial propagation in the hyperbolic space as follows:

$$\mathbf{p}^{t+1}_{(x,y)} = A_{(x,y)} \odot \mathbf{p}^{0}_{(x,y)} + \sum_{(l,m)\in\mathcal{N}_{(x,y)}} A_{(l,m)} \odot \mathbf{p}^{t}_{(l,m)}, \quad (15)$$

where A is an affinity map that is back-projected to the Euclidean space from the hyperbolic feature map $\widetilde{\mathbf{H}}$. The $\odot$ operator represents an element-wise product, and $\mathcal{N}_{(x,y)}$ indicates the locations of neighbor pixels in $\mathbf{p}$. We position HAM at the end of the affinity branch in the SPNs (Fig. 4), where the convolutional features in the top layer have a more hierarchical property than the features at the bottom layers.

5. Experiments

We conduct a variety of experiments on spatial propagation tasks, including depth completion (Sec. 5.1) and semantic segmentation (Sec. 5.2) as shown in Fig. 4. Moreover, we provide ablation studies to describe the effects of each component in HAM and the robustness of our method concerning input sparsity and feature compression (Sec. 5.3). Note that we describe details about the experimental setup, more quantitative and qualitative results, and further analysis in the supplemental materials.

5.1. Depth Completion

Given an RGB image and a sparse depth map (*e.g.,* point cloud), the depth completion task produces a dense depth map at a camera viewpoint (Fig. 4 (a)). From this problem definition, the sparse depth samples can be regarded as given seeds, and SPNs are trained to infer proper pixel affinities to propagate depth values into entire pixels. A pioneering study, Convolutional Spatial Propagation Network (CSPN) (Cheng et al., 2018), treats sparse depth samples as seeds and infers affinity maps to operate N iterations of spatial propagation. We select this fully convolutional model as a baseline and replace convolutional layers in the affinity branch with our HAM. With this baseline, we conduct the depth completion task on the NYUv2, the ScanNet, and the Virtual-KITTIv2. Note that we follow the original training scheme in (Cheng et al., 2018) for a fair comparison.

We conduct our experiment on the NYUv2 dataset (Silberman et al., 2012) which provides RGB images and dense depth pairs for 464 indoor scenes captured from RGB-D sensors. Using an official train/test split, we generate random depth samples as proposed in the baseline model, as n Fig. 5. We compare the baseline model (CSPN) and naïve hyperbolic approach with our HAM using official evaluation metrics (Eigen et al., 2014; Uhrig et al., 2017)[1]: RMSE, MAE, iRMSE, iMAE, REL, and $\delta^1_{1.25}$. As shown in Tab. 2, our method outperforms the baseline as well as the basic hyperbolic convolution operation. The strength of our method is qualitatively supported by Fig. 5. The results show that HAM preserves 3D shapes of slanted objects, homogeneous surfaces, and thin structures well. According to our analysis of pixel embeddings in Fig. 7-(a), we deduce pixel affinities

[1]The details are described in our supplementary material.

Table 3. Quantitative semantic segmentation results on PASCAL VOC 2012 (Everingham et al., 2015). Following (Jiang et al., 2018), we set simplified DeeplabV2 (Chen et al., 2018) models as our baseline. (**Bold**: Best, Underline: Second Best)

	Semantic Segmentation (Unit: mIoU, **Bold**: Best, Underline: Second Best)																					
	plane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	TV	mIoU	Pix-Acc
Sim-Deeplab	79.37	35.94	76.15	59.38	65.21	82.15	76.67	81.82	27.05	68.79	47.99	75.10	67.83	74.66	78.46	48.56	70.49	40.47	80.76	68.47	66.39	90.06
DifNet	82.97	37.36	82.82	52.98	72.63	86.56	79.76	87.99	28.43	75.87	51.81	80.67	79.61	75.00	82.77	53.69	77.18	40.75	83.15	68.18	70.02	91.58
Naïve	86.09	38.40	79.71	54.92	68.68	85.09	81.89	88.05	35.38	78.84	47.35	79.54	77.46	78.08	82.88	50.94	82.71	40.35	80.00	73.96	70.49	91.53
Ours	84.90	37.89	82.02	60.61	67.82	85.13	83.41	87.40	36.37	81.12	47.85	78.54	77.30	78.52	81.26	57.73	81.38	42.56	81.37	71.19	**71.16**	**91.61**

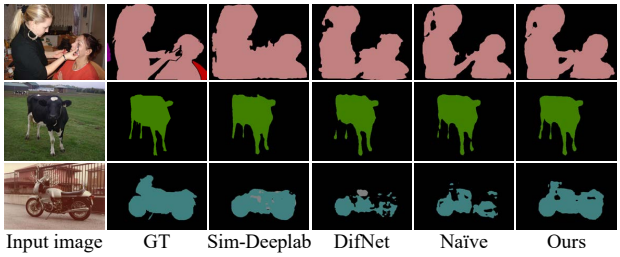


Figure 6. Qualitative results on Pascal VOC 2012 dataset (Everingham et al., 2015). It demonstrates our HAM yields high accuracy in the estimation of thin structures and objects' boundaries, which exhibits a similar tendency to the depth completion task.

Table 4. Quantitative semantic segmentation results on various datasets. (**Bold**: Best, Underline: Second Best)

	Pascal Context		NYUv2		ADE20K	
	mIoU	Pix-Acc	mIoU	Pix-Acc	mIoU	Pix-Acc
Sim-Deeplab	57.12	72.69	28.42	57.09	21.69	60.09
DifNet	59.77	74.19	28.91	57.48	23.30	63.21
Naïve	60.18	74.51	28.19	56.78	23.44	63.92
Ours	**60.30**	**74.81**	**30.45**	**58.97**	**25.28**	**63.94**

are well clustered along with their hierarchy in hyperbolic space, regardless of their shapes in Euclidean space.

Virtual-KITTIv2 (Cabon et al., 2020) is a photo-realistic synthetic dataset with a dense depth map, and ScanNet (Dai et al., 2017) provides images and depth maps in indoor environments. We split the virtual-KITTIv2 into one scene (Scene-02) for validation and the other scenes (Scene-01,06,18,20) for training without any temporal overlap between them. For ScanNet, we follow the official train/test split: 1,513 scenes for training and 100 scenes for the test. We comply with the authors' guideline of (Cheng et al., 2018) for depth sampling on both datasets for both training and evaluation. Tab. 2 turns out that HAM consistently outperforms the comparison methods in almost metrics without additional learnable parameters. Thanks to representation power and spatially-varying weights based on hyperbolic geometry, there are significant margins in both datasets (about 10% lower RMSE) over the baseline method.

5.2. Semantic Segmentation

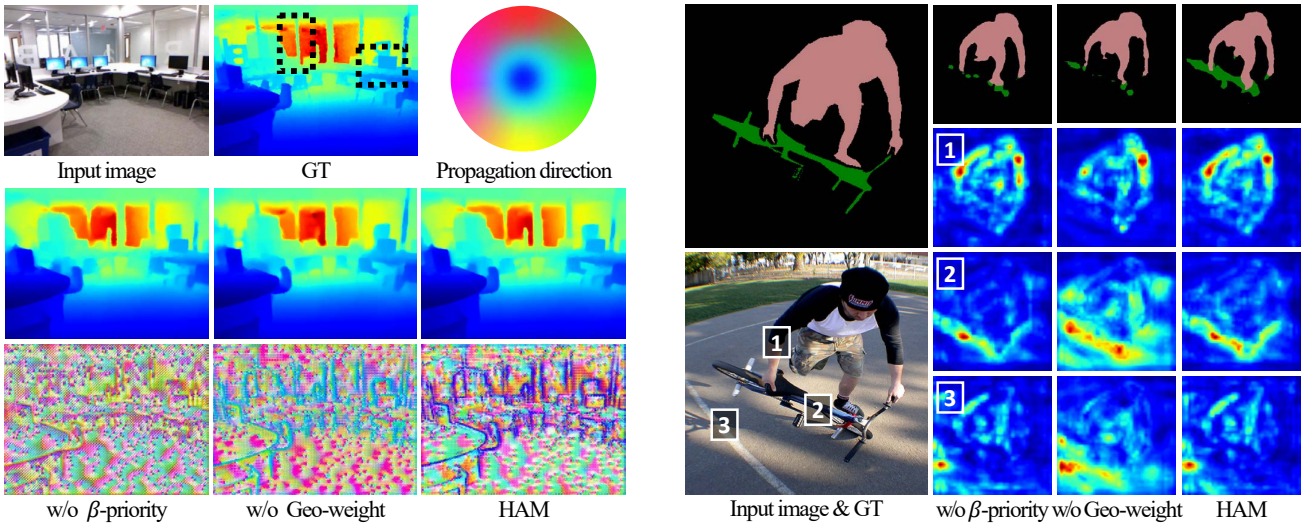
Semantic segmentation aims to perform pixel-wise classification, which can be formulated as spatial propagation. In terms of spatial propagation, the semantic segmentation needs to infer both an initial estimate and a similarity map as an affinity, as illustrated in Fig. 4 (b).

DifNet (Jiang et al., 2018) presents a network that utilizes a cascade of random walks to approximate a diffusion process. It initially detects seed pixels from an input image and computes similarity maps, and then propagates the seed information to the whole semantic map along with the estimated similarity maps. In this work, we select it as a baseline network for our semantic segmentation task. For a fair comparison, we use an identical backbone network (Chen et al., 2018) and take the same propagation process (*i.e.,* 5 random walks with 5 transition matrices for final estimation).

Augmented Pascal VOC 2012 (Everingham et al., 2015) dataset provides 10,582 training, 1,449 validation, and 1,456 test images with pixel-level labels in 20 foreground object classes and one background class. As shown in Tab. 3, our HAM outperforms the baseline as well as the naïve approach. The performance of the naïve is even worse than that of the baseline method working in Euclidean space regarding pixel accuracy. In particular, the qualitative results from our HAM in Fig. 6 yield high accuracy when estimating thin structures and objects' boundaries. This tendency seems similar to the depth completion task in Sec. 5.1. We will further describe the fundamental reasons for this tendency regarding affinity computation in Sec. 5.3.

We validate the scalability of our method on larger datasets, *i.e.,* NYUv2 (Silberman et al., 2012), Pascal-Context (Gupta et al., 2013), and ADE20K (Zhou et al., 2017). Following (Gupta et al., 2013), we conduct a semantic segmentation experiment on NYUv2 dataset (Silberman et al., 2012) which provides 1,449 images and 40 category object labels. We also report results on the PASCAL-Context dataset with 10,103 images. We use 34 object classes provided by (Mottaghi et al., 2014) and ResNet50 as the backbone to train all methods for 200 epochs. ADE20K contains 22,210 images from 150 classes and is split into 20,210 in the train set and 2000 images in the test set.

As reported in Tab. 4, our HAM outperforms the comparison methods, even with the increasing number of semantic labels. It is particularly notable that the performance difference between ours and naïve on NYUv2 and ADE20K is about 2% without additional learnable parameters. We observe that HAM is effective for thin-structure object.



(a) Depth Completion (b) Semantic Segmentation

Figure 7. Affinity analysis. (a) Comparison of affinity maps where HAM captures boundary information well. Note that we use a Principle Component Analysis (Jolliffe, 1986) for visualization. (b) Similarity maps visualization under three selected nodes (1 : person, 2 : bicycle, and 3 : background). Nodes with more red highlights are more similar to the selected node.

Table 5. Ablation study on NYUv2 (Silberman et al., 2012) and PASCAL VOC 2012 datasets (Everingham et al., 2015). We report IoUs for selected classes to highlight the effectiveness of each component in HAM. The mIoU represents the average over all the classes of PASCAL VOC 2012 dataset. (**Bold**: Best)

	Depth Completion (Unit: meter, **Bold**: Best)							Semantic Segmentation (Unit: mIoU, **Bold**: Best)							
	RMSE	MAE	iRMSE	iMAE	REL	$\delta^1_{1.25}$	$\delta^2_{1.25}$	boat	bottle	chair	cow	mbike	plant	train	mIoU
w/o β-priority	0.1047	0.0403	**0.0158**	**0.0059**	0.0137	**0.9942**	0.9990	58.57	65.42	34.72	77.84	76.79	55.67	79.37	70.77
w/o Geo-weight	0.1050	0.0409	0.0162	0.0060	0.0140	0.9940	0.9990	60.26	64.85	32.07	78.98	75.83	56.16	80.23	70.91
Ours	**0.1022**	**0.0395**	**0.0158**	**0.0059**	**0.0136**	**0.9942**	**0.9991**	**60.61**	**67.82**	**36.37**	**81.12**	**78.52**	**57.73**	**81.37**	**71.16**

Table 6. Quantification analysis of Fig. 7. To quantify each affinity and similarity map, we calculate the propagation cost calculated with Eq (16) based on the ground truth and the inferenced affinity of pixels on the propagation path.

Method	Depth Completion	Semantic Segmentation
w/o β-priority	0.85	0.31
w/o Geo-weight	0.81	0.33
Ours	**0.76**	**0.25**

5.3. Ablation Study

β-priority and geodesic weight. We conduct an ablation study of HAM by intentionally omitting each component, β-priority, and geodesic weight. As shown in Tab. 5 and Fig. 7, both β-priority and geodesic weight consistently improve the performances of depth completion as well as semantic segmentation. We observe that β-priority is robust in capturing thin structures due to their priority-based concatenation scheme that selects the most correlated signals. Furthermore, in Fig. 7-(b), the geodesic weight emphasizes the correlated attentions (red) and suppresses the redundant similarities (blue), which is beneficial to cope with boundary ambiguities in the semantic segmentation. Additionally, to verify the numerical results of Fig. 7, we quantify each affinity and similarity map in the following steps:

1. Sample pixels (*i.e.,* seed) at sparse depth or initial semantic and corresponding pixels at a certain distance.
2. Calculate a *Propagation Cost* between the ground truth D_{gt} of the sampled pixels and the mean affinity A_{mean} on the propagation path as follows[2]:

$$\text{Propagation Cost} = cos(\frac{\pi}{2}A_{\text{mean}} - \frac{\pi}{20}D_{\text{gt}}). \quad (16)$$

3. Iterate 1000 times for (1) and (2) and average the calculated cost for spatial propagation in the scene.

By doing this, we can compare the cost for propagation between two points. As shown in Tab. 6, HAM has the lowest value, which indicates it constructs the most high-fidelity relationship between pixels and the most accurate affinity and similarity map.

[2]Since the ranges of mean affinity and ground truth for the NYUv2 depth dataset are $0 \leq A_{\text{mean}} \leq 1$ and $0 \leq D_{\text{gt}} \leq 10$, respectively, we normalize each value and utilize a periodic function to make the cost value from 0 to 1.

Analysis of affinity and similarity map. We provide further analysis of our HAM with respect to the affinity and similarity map. In the depth completion network, we follow the visualization scheme in (Tang et al., 2020) to depict the affinity map (*i.e.,* the direction of pixel-wise propagation). As shown in Fig. 7-(a), our HAM captures the depth boundaries well where usually uncertain in the original propagation approaches, resulting in distinctive propagation.

In the semantic segmentation task, a similarity map represents a pixel-wise feature similarity that guides the diffusion of initial seeds in a coarse-to-fine manner. According to Fig. 7-(b), the inferred similarity maps using HAM can identify fine-grained affinities among pixels. It also demonstrates that our method can capture long-range connections (node 1) and achieve robustness in identifying thin structures (node 2) as well as backgrounds (node 3). Based on the observations, we argue that HAM takes advantage of the hierarchical representation from pixel relation priority and the spatial attention from the components, resulting in reliable predictions at thin structures and sharp boundaries. More examples are reported in the supplementary materials.

Sparsity and low-dimensional embedding. Existing hyperbolic embeddings (Nickel & Kiela, 2018; 2017) show their usefulness in harsh condition, such as low-dimensional setup. To check the robustness of input sparsity for the depth completion task, we randomly drop depth points in ground-truth depth maps with different ratios. We also tune the number of iterations to converge the network toward minimum errors. The results in Fig. 8-(a) support the claim for the sparsity of input.

As shown in Fig. 8-(b), we set the different numbers of channel dimensions in Euclidean features which are input to our HAM. As the number of the dimensions decreases, the performance notably drops in the baseline model (Jiang et al., 2018). However, the performance drop in our network is smaller than that of other methods, including the naïve hyperbolic network. It demonstrates that HAM still produces accurate similarity maps for spatial propagation.

Various spatial propagation schemes. Next, we apply our HAM into different types of depth completion methods: NLSPN (Park et al., 2020) and DYSPN (Lin et al., 2022)[3]. The models follow a similar propagation scheme with CSPN, constructing affinities with standard Euclidean CNNs. The difference between CSPN and them is to adopt the non-local spatial propagation with the deformable convolution and the non-linear spatial propagation based on the dynamic convolution filter, respectively. For these models, we also replace the affinity branch with our HAM.

As shown in Fig. 8-(a) again, the application of HAM to all the methods consistently shows significant performance gains, especially with sparser measurements (*e.g.,* 100 and 250). This indicates that the HAM synergies well with the trendy baselines. The reason for the noticeable gain by HAM is that the limited offset of the deformable convolution in NLSPN and the receptive fields in the dynamic attention module of DYSPN cannot capture the long-range connection among pixels. In addition, we point out that results from the trendy models tend to be highly engineered for the specific setup (i.e., 500 points as initial seeds) for the benchmark result, which sometimes reveals the weakness of the sparser setups like 16- and 32-line LiDARs. Therefore, we argue that HAM is beneficial for real-world scenarios.

[3]Since no public codes are available, we implement DYSPN ourselves with the almost similar performance of (Lin et al., 2022).

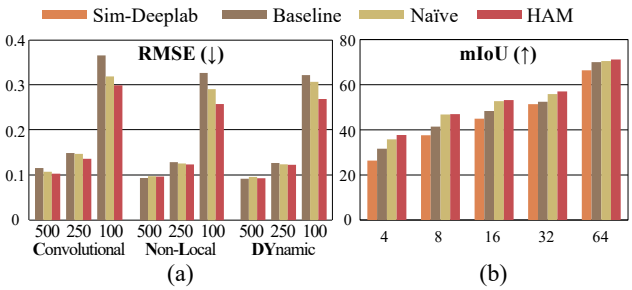


Figure 8. Sparsity and low-dimensional embedding. (a) Depth completion comparison results for various spatial propagation schemes (**C**onvolutional (**C**-SPN), **N**on-**L**ocal (**NL**-SPN), and **DY**namic (**DY**-SPN)) w.r.t. the number of samples. (b) Semantic segmentation results w.r.t. the number of dimensions to generate similarity maps.

6. Conclusion

We present a Hyperbolic Affinity learning Module (HAM) for robust spatial propagation. HAM is a differentiable layer with a β-priority and a geodesic weight and is easily incorporated into conventional SPNs. Though HAM does not require extra learnable parameters over baseline methods, HAM achieves promising results for depth completion and semantic segmentation on various datasets.

Acknowledgements

This research was paritially supported by the National Research Foundation of Korea (NRF) (No.2020R1C1C1012635) grant, the Institute of Information & communications Technology Planning & Evaluation (IITP) (No.2019-0-01842, Artificial Intelligence Graduate School Program (GIST), No.2021-0-02068, Artificial Intelligence Innovation Hub) grants funded by the Korea government(MSIT), 'Project for Science and Technology Opens the Future of the Region' program through the INNOPOLIS FOUNDATION funded by Ministry of Science and ICT (Project Number: 2022-DD-UP-0312), and GIST-MIT Research Collaboration funded by the GIST.

References

Bao, L., Song, Y., Yang, Q., Yuan, H., and Wang, G. Tree filtering: Efficient structure-preserving smoothing with a minimum spanning tree. *IEEE Transactions on Image Processing*, 23(2):555–569, 2013.

Becigneul, G. and Ganea, O.-E. Riemannian adaptive optimization methods. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2019.

Bertasius, G., Torresani, L., Yu, S. X., and Shi, J. Convolutional random walk networks for semantic image segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Bridson, M. R. and Haefliger, A. *Metric Spaces of Non-Positive Curvature*. Springer, 1999.

Cabon, Y., Murray, N., and Humenberger, M. Virtual kitti 2. In *arXiv*, 2020.

Caesar, H., Uijlings, J., and Ferrari, V. Coco-stuff: Thing and stuff classes in context. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Chami, I., Ying, R., Ré, C., and Leskovec, J. Hyperbolic graph convolutional neural networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2019.

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 (PAMI)*, 2018.

Cheng, X., Wang, P., and Yang, R. Depth estimation via affinity learned with convolutional spatial propagation network. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2018.

Cheng, X., Wang, P., and Yang, R. Learning depth with convolutional spatial propagation network. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2019.

Cheng, X., Wang, P., Guan, C., and Yang, R. Cspn++: Learning context and resource aware convolutional spatial propagation networks for depth completion. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

Choe, J., Joo, K., Imtiaz, T., and Kweon, I. S. Volumetric propagation network: Stereo-lidar fusion for long-range depth estimation. *IEEE Robotics and Automation Letters*, 6(3):4672–4679, 2021.

Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., and Niessner, M. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Dai, J., Wu, Y., Gao, Z., and Jia, Y. A hyperbolic-to-hyperbolic graph convolutional network. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Dai, L., Yuan, M., Zhang, F., and Zhang, X. Fully connected guided image filtering. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 352–360, 2015.

Eigen, D., Puhrsch, C., and Fergus, R. Depth map prediction from a single image using a multi-scale deep network. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2014.

Ermolov, A., Mirvakhabova, L., Khrulkov, V., Sebe, N., and Oseledets, I. Hyperbolic vision transformers: Combining improvements in metric learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. doi: 10.48550/ARXIV.2203.10833. URL https://arxiv.org/abs/2203.10833.

Everingham, M., Eslami, S. M., Gool, L., Williams, C. K., Winn, J., and Zisserman, A. The pascal visual object classes challenge: A retrospective. *International Journal on Computer Vision (IJCV)*, 2015.

Farbman, Z., Fattal, R., and Lischinski, D. Diffusion maps for edge-aware image editing. *ACM Transactions on Graphics (TOG)*, 2010.

Fournier, H., Ismail, A., and Vigneron, A. Computing the gromov hyperbolicity of a discrete metric space. *Information Processing Letters*, 115(6):576–579, 2015. ISSN 0020-0190. doi: https://doi.org/10.1016/j.ipl.2015.02.002. URL https://www.sciencedirect.com/science/article/pii/S0020019015000198.

Ganea, O., Becigneul, G., and Hofmann, T. Hyperbolic neural networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2018.

Gao, Z., Wu, Y., Jia, Y., and Harandi, M. Curvature generation in curved spaces for few-shot learning. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2021.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research (IJRR)*, 32(11):1231–1237, 2013.

GhadimiAtigh, M., Schoep, J., Acar, E., van Noord, N., and Mettes, P. Hyperbolic image segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. doi: 10.48550/ARXIV.2203.05898. URL https://arxiv.org/abs/2203.05898.

Gromov, M. *Hyperbolic Groups*. Springer New York, 1987.

Gulcehre, C., Denil, M., Malinowski, M., Razavi, A., Pascanu, R., Hermann, K. M., Battaglia, P., Bapst, V., Raposo, D., Santoro, A., and de Freitas, N. Hyperbolic attention networks. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2019.

Gupta, S., Arbelaez, P., and Malik, J. Perceptual organization and recognition of indoor scenes from rgb-d images. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

Hsu, J., Gu, J., Wu, G., Chiu, W., and Yeung, S. Capturing implicit hierarchical structure in 3d biomedical images with self-supervised hyperbolic representations. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2021.

Jiang, P., Gu, F., Wang, Y., Tu, C., and Chen, B. Difnet: Semantic segmentation by diffusion networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2018.

Jolliffe, I. *Principal Component Analysis*. Springer Verlag, 1986.

Khrulkov, V., Mirvakhabova, L., Ustinova, E., Oseledets, I., and Lempitsky, V. Hyperbolic image embeddings. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2015.

Krioukov, D., Papadopoulos, F., Kitsak, M., Vahdat, A., and Bogu, M. Hyperbolic geometry of complex networks. *Physical Review E*, 2010.

Levin, A., Lischinski, D., and Weiss, Y. Colorization using optimization. In *Proceedings of ACM SIGGRAPH*, 2004.

Lin, Y., Cheng, T., Zhong, Q., Zhou, W., and Yang, H. Dynamic spatial propagation network for depth completion. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2022.

Linial, N., London, E., and Rabinovich, Y. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, 1995.

Liu, Q., Nickel, M., and Kiela, D. Hyperbolic graph neural networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2019.

Liu, S., De Mello, S., Gu, J., Zhong, G., Yang, M.-H., and Kautz, J. Learning affinity via spatial propagation networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2017.

Liu, S., Chen, J., Pan, L., Ngo, C.-W., Chua, T.-S., and Jiang, Y.-G. Hyperbolic visual embedding learning for zero-shot recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Long, J., Shelhamer, E., and Darrell, T. Fully convolutional networks for semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Long, T., Mettes, P., Shen, H. T., and Snoek, C. G. M. Searching for actions on the hyperbole. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Ma, R., Fang, P., Drummond, T., and Harandi, M. Adaptive poincaré point to set distance for few-shot classification. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2022.

Maire, M., Narihira, T., and Yu, S. X. Affinity cnn: Learning pixel-centric pairwise relations for figure/ground embedding. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

McInnes, L., Healy, J., Saul, N., and Großberger, L. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 2018. doi: 10.21105/joss.00861. URL https://doi.org/10.21105/joss.00861.

Mostajabi, M., Yadollahpour, P., and Shakhnarovich, G. Feedforward semantic segmentation with zoom-out features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Mottaghi, R., Chen, X., Liu, X., Cho, N.-G., Lee, S.-W., Fidler, S., Urtasun, R., and Yuille, A. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Nickel, M. and Kiela, D. Poincaré embeddings for learning hierarchical representations. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2017.

Nickel, M. and Kiela, D. Learning continuous hierarchies in the Lorentz model of hyperbolic geometry. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018.

Park, J., Joo, K., Hu, Z., Liu, C.-K., and Kweon, I. S. Non-local spatial propagation network for depth completion. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2020.

Ryohei, S., Yusuke, M., and Tatsuya, H. Hyperbolic neural networks++. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2021.

Sala, F., De Sa, C., Gu, A., and Re, C. Representation tradeoffs for hyperbolic embeddings. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018.

Sarkar, R. Low distortion delaunay embedding of trees in hyperbolic plane. In *International Symposium on Graph Drawing and Network Visualization*, 2012.

Shi, J. and Malik, J. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2000.

Shin, J., Shin, S., and Jeon, H.-G. Task-specific scene structure representations. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2023.

Silberman, N., Hoiem, D., Kohli, P., and Fergus, R. Indoor segmentation and support inference from rgbd images. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2012.

Tang, J., Tian, F.-P., Feng, W., Li, J., and Tan, P. Learning guided convolutional network for depth completion. *IEEE Transactions on Image Processing (TIP)*, 2020.

Tifrea, A., Bécigneul, G., and Ganea, O.-E. Poincaré glove: Hyperbolic word embeddings. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2019.

Tu, W.-C., He, S., Yang, Q., and Chien, S.-Y. Real-time salient object detection with a minimum spanning tree. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2334–2342, 2016.

Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., and Geiger, A. Sparsity invariant cnns. In *International Conference on 3D Vision (3DV)*, 2017.

Ungar, A. A. Hyperbolic trigonometry and its application in the poincaré ball model of hyperbolic geometry. *Computers & Mathematics with Applications*, 41(1-2):135–147, 2001.

Ungar, A. A. A gyrovector space approach to hyperbolic geometry. In *Synthesis Lectures on Mathematics and Statistics*, 2008.

Wang, X., Girshick, R., Gupta, A., and He, K. Non-local neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Weickert, J. *Anisotropic diffusion in image processing*. Teubner Stuttgart, 1998.

Yan, J., Luo, L., Deng, C., and Huang, H. Unsupervised hyperbolic metric learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Yang, Q. A non-local cost aggregation method for stereo matching. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

Yang, Q. Stereo matching using tree filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37 (4):834–846, 2014.

Yatziv, L. and Sapiro, G. Fast image and video colorization using chrominance blending. *IEEE Transactions on Image Processing (TIP)*, 2006.

Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2014.

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. Scene parsing through ade20k dataset. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Supplementary Material for Learning Affinity with Hyperbolic Representation for Spatial Propagation

In this supplementary material, we first describe details of implementation and evaluation metrics used in our main manuscript (Appendix A). We then provide quantification analysis on hyperbolicity that indicates a tree-likeliness of embedded pixel features (Appendix C) on 2D Poincaré ball (Appendix B). We also introduce a toy example to check the long-range connection (Appendix D) and carry out additional experiments on the depth completion task for the KITTI Depth Completion (KITTI-DC) dataset (Uhrig et al., 2017) that provides higher resolution outdoor images than the NYUv2 dataset (Appendix E). Moreover, we conduct an additional experiment on a large dataset, COCO-Stuff10K (Caesar et al., 2018) that involves a more complex semantic object configuration than that of other datasets as described in the manuscript (Appendix F). Finally, we represent extensive results evaluations and their examples on NYUv2 dataset (Silberman et al., 2012), Pascal VOC 2012 dataset (Everingham et al., 2015) (Figure F M and Table D F).

A. Experimental Details

A.1. Implementation Details

For implementation of our operation HAM, it requires several hyper-parameters to properly transform Euclidean features $\mathbf{f}$ into hyperbolic features $\mathbf{h}$ and convolution them. We set the curvature κ to 0.1 and a size of the kernel γ to 3, and utilize a 2-dimensional hyperbolic convolution operation for spatial propagation tasks. We optimize our methods using Adam optimizer (Kingma & Ba, 2015) with $\beta 1 = 0.9$, $\beta 2 = 0.999$ with the initial learning rate of 0.001, and it takes about 1 day for training networks using 4 NVIDIA RTX 3090 GPU. According to (Chami et al., 2019), we observe that Euclidean optimization is substantially more stable than Riemannian optimization (Becigneul & Ganea, 2019). Due to the exponential and logarithmic maps which are mapping functions between Euclidean space and hyperbolic space, the parameters of hyperbolic model can be optimized using Euclidean optimization.

A.2. Evaluation Metrics: Depth Completion

We introduce a depth quality evaluation metrics, proposed in (Eigen et al., 2014; Geiger et al., 2013; Uhrig et al., 2017). Given a ground truth depth $D = \{d\}$ and the predicted depth $\hat{D} = \{\hat{d}\}$, the metrics are as follows:

- RMSE: Root mean squared error. $\sqrt{\frac{1}{|D|}\sum_{\hat{d}\in\hat{D}}|\hat{d}-d|^2}$
- MAE: Mean absolute error. $\frac{1}{|D|}\sum_{\hat{d}\in\hat{D}}|\hat{d}-d|$
- iRMSE: Root mean squared error of the inverse depth. $\sqrt{\frac{1}{|D|}\sum_{\hat{d}\in\hat{D}}|\frac{1}{\hat{d}}-\frac{1}{d}|^2}$
- iMAE: Mean absolute error of the inverse depth. $\frac{1}{|D|}\sum_{\hat{d}\in\hat{D}}|\frac{1}{\hat{d}}-\frac{1}{d}|$
- REL: Mean absolute relative error. $\frac{1}{|D|}\sum_{\hat{d}\in\hat{D}}|\frac{\hat{d}-d}{d}|$
- δ_i : percentage of predicted pixels where the relative error is within a threshold.

$$\delta_i = \frac{card\left(\left\{\hat{d}\in\hat{D} : \max\left\{\frac{\hat{d}}{d}, \frac{d}{\hat{d}}\right\} < 1.25^i\right\}\right)}{card\,(D)}$$

where the $card$ is the cardinality of a set. A higher δ_i indicates better prediction.

A.3. Evaluation Metrics: Semantic Segmentation

We provide details of evaluation metrics for semantic segmentation. Let x_{ik} be the number of predicted pixels as a class i. Here, its ground-truth class denotes k whose number is N_{class}. The total number of ground-truth pixels for all classes

is $T_i = \sum_k x_{ik}$. In this work, we use standard metrics (Jiang et al., 2018; Mostajabi et al., 2015; Long et al., 2015) for category-level segmentation including a pixel accuracy and a region intersection over union (IoU) as below:

- Pixel Accuracy:

$$\frac{1}{N_{class}} \sum_i \frac{x_{ii}}{T_i}$$

- Mean IoU:

$$\frac{1}{N_{class}} \sum_i \frac{x_{ii}}{T_i + \sum_j x_{ji} - x_{ii}}$$

.

B. Affinity Feature Embedding on 2D Poincaré Ball

In Figure 1 of the manuscript, we provide our own analysis of the feature embedding from ours and previous studies. For more analysis, we provide the further details of pixel feature embedding on the 2D Poincaré ball. It shows interesting pixel embedding distributions that result in the accurate spatial propagation operation under ambiguous regions.

As shown in Fig. 9, we show both the spatial propagation process of all methods (baseline (Cheng et al., 2018), naïve (Ryohei et al., 2021), and HAM) and pixel patch embeddings to analyze the causality of the different results. We can verify the occurrences of bleeding error, which results from the ambiguity of boundary information as demonstrated in Fig. 13.

In Fig. 10, we provide additional feature embeddings on the Poincaré ball model and visualization of the corresponding affinity maps that are omitted due to the space limit in the manuscript. Visually, it shows that well-separated foreground, background, and boundary pixels can be achievable from ours, whereas not from the baseline and the naïve approach.

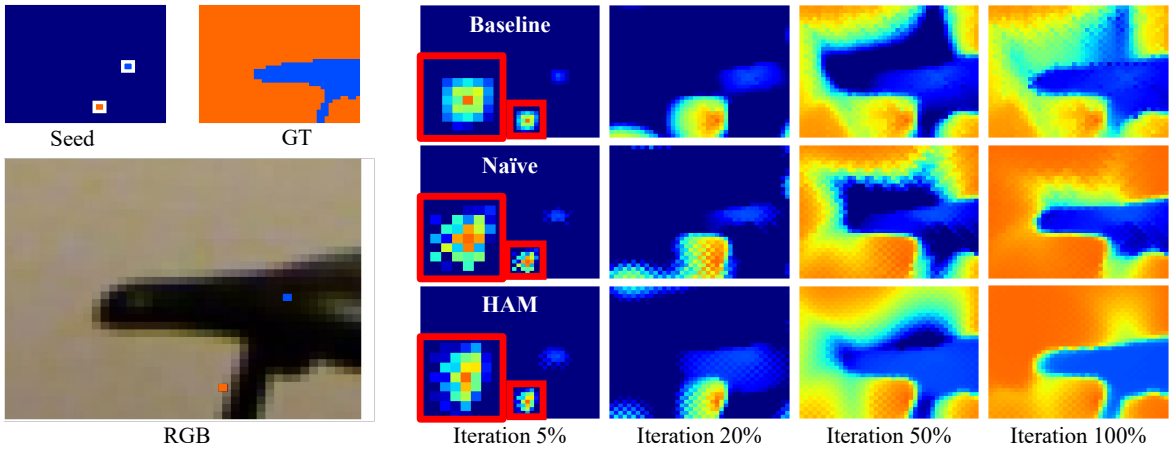


Figure 9. Example of spatial propagation from a conventional method (Cheng et al., 2018), naïve approach and our HAM given two initial seeds and a RGB image.

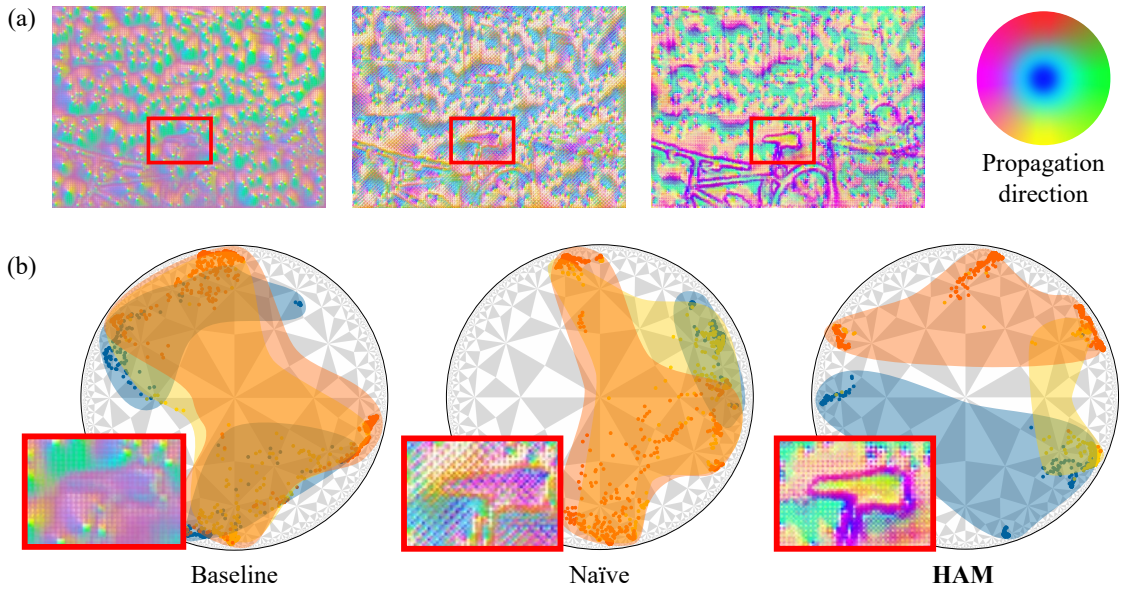


Figure 10. (a) Affinity visualization as an extension of Figure 1 of the manuscript. Note that we use a Principle Component Analysis (Jolliffe, 1986) for visualization. (b) Visualizing the pixel feature embedding on 2D Poincaré ball using UMAP (McInnes et al., 2018) based on hyperboloid distance. We embed learned affinity features and inferred depth maps on 2D Poincaré ball.

C. Measure of Data Hyperbolicity: δ-hyperbolicity

Table 7. Calculate relative δ-hyperbolicity (δ_{hyp}) on various dataset (Silberman et al., 2012; Cabon et al., 2020; Dai et al., 2017; Geiger et al., 2013; Gupta et al., 2013; Everingham et al., 2015; Zhou et al., 2017) using learned features to construct affinity or similarity map within SPNs((Cheng et al., 2018; Jiang et al., 2018). Lower Values of δ_{hyp} indicate a higher degree of data hyperbolicity. Results are averaged across testset of size 1000 and the standard deviation for all the experiments did not exceed 0.002.

	Depth Completion				Semantic Segmentation			
	NYUv2	KITTI	vKITTIv2	ScanNet	Pascal-VOC	NYUv2	Pascal-Context	ADE20K
δ_{hyp}	0.326	0.263	0.275	0.271	0.237	0.226	0.198	0.259

A concept of hyperbolicity is used to measure a "tree-likeness" of a graph in terms of distance metric. This metric is based on the concept of Gromov δ-hyperbolicity (Bridson & Haefliger, 1999; Gromov, 1987), which captures fundamental characteristics of negatively curved spaces such as the hyperbolic space and discrete spaces such as trees. A low δ-hyperbolicity denotes that a set has an underlying hyperbolic geometry, *i.e.,* it is an approximately tree-like structure. Conversely, a high δ-hyperbolicity suggests that it obtains long cycles, or could not be embedded in a low dimensional hyperbolic space without distortion. For instance, the Euclidean space $\mathbb{R}^n$ is ∞-hyperbolic, while the standard Poincaré disk $\mathbb{D}^2$ is known to have a δ-hyperbolicity of $\log(1+\sqrt{2}) \simeq 0.88$ (Tifrea et al., 2019). Here, we can calculate the δ-hyperbolicity for an arbitrate set in the following manner: Let $\mathcal{X}$ be an arbitrary metric space endowed with the distance function d, and the *Gromov product* (Gromov, 1987) for points $x, y, z \in \mathcal{X}$ is defined as:

$$(y, z)_x = \frac{1}{2}(d(x, y) + d(x, z) - d(y, z)). \tag{17}$$

Then, the smallest value such that the following four-point condition holds for all points is defined as δ as follows:

$$(x, z)_w \geq \min\left((x, y)_w, (y, z)_w\right) - \delta. \tag{18}$$

For computing the δ-hyperbolicity, we follow an efficient method presented in (Fournier et al., 2015): For a set of points,

we compute the matrix M of a pairwise Gromov product. The δ value is then defined as the largest entry in the matrix $(M \otimes M) - M$. Here, $\otimes$ indicates the min-max matrix product defined as $(A \otimes B)_{ij} = \max_k \min\{A_{ik}, B_{kj}\}$.

In order to verify the validity to use hyperbolic representation for affinity construction in spatial propagation tasks, we adopt the procedure described in (Khrulkov et al., 2020; Ermolov et al., 2022), so called relative δ hyperbolicity which is defined as $\delta_{hyp}(X) = 2\delta(X)/\mathrm{diam}(X)$, where $\mathrm{diam}(X)$ denotes a diameter of the set. To prove its validity and efficacy of the hyperbolic embedding of the feature map extracted from images, we follow the quantitative analysis described in (Khrulkov et al., 2020; Ermolov et al., 2022). They adopt a relative δ hyperbolicity, defined as $\delta_{hyp}(X) = 2\delta(X)/\mathrm{diam}(X)$ where $\mathrm{diam}(X)$ denotes a diameter of a set. The relative δ hyperbolicity means how close the dataset is to hyperbolic. As shown in Tab. 7, the δ_{hyp} are significantly closer to 0. Since the δ_{hyp} values for metric learning (Ermolov et al., 2022) and few-shot learning (Khrulkov et al., 2020), which are computed based on ImageNet-pretrained features extracted from standard feature extractors (*e.g.,* VGG, ResNet, and visual transformers), are mostly 0.2∼0.4, our choice to apply hyperbolic embedding into the spatial propagation tasks is also reasonable. Note that we measure the relative δ hyperbolicity using the feature map from the original SPNs whose backbone is ImageNet-pretrained ResNet. Through our study of hyperbolicity, it seems to be an appropriate choice to apply hyperbolic embedding into pixel affinity construction when spatial propagation.

D. Toy Example : Long-Range Connectivity

Table 8. Quantitative results on spatial corruptions. Parentheses mean performance gaps over the normal conditions (Table 1 and 2 of the manuscript).

	Depth Completion		Semantic Segmentation
	RMSE	MAE	mIoU
Baseline	0.2162 (+0.1003)	0.0896 (+0.0421)	65.80% (-4.22%)
Naïve	0.1518 (+0.0442)	0.0594 (+0.0163)	66.67% (-3.82%)
Ours	0.1420 (+0.0387)	0.0547 (+0.0149)	67.42% (-3.54%)

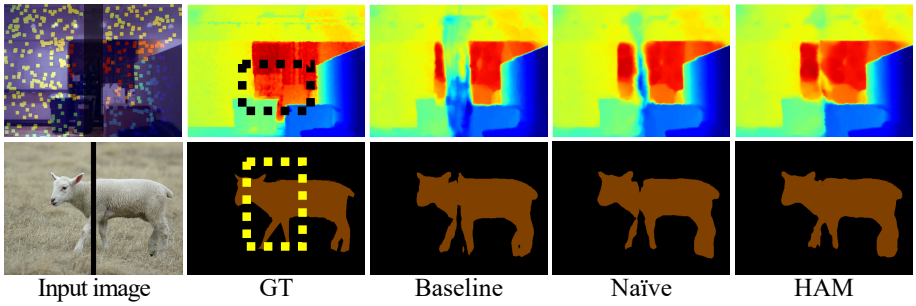


Figure 11. Qualitative results on long-range connectivity.

For more straightforward understanding of the long-range connectivity through our HAM, we design new experimental setups. As shown in Figure Fig. 11, we intentionally mask out the part of depth (224(height) × 24(width) pixels) to impose affinity uncertainty for the depth completion task. Similarly, we also mask out the center regions in the semantic segmentation task (321(height) × 9(width) pixels). By omitting the part of the seeds, we test the ability of long-range spatial propagation from previous studies and ours.

As in shown in the Tab. 8 and Fig. 11, it turns out that our method demonstrates less performance drop over the normal condition (Table 1 and 2 of the manuscript). For each task, HAM achieves the better performance than the accuracy of both the baseline networks (Jiang et al., 2018; Cheng et al., 2018) and the naïve method (Ryohei et al., 2021). Both the naïve and HAM approaches construct affinities well due to the merit of the hyperbolic embedding, even between separated pixels in Euclidean space. In particular, our HAM shows better performance than the naïve method since the geodesic-awareness enforces the selection of semantic features for the same classes.

Table 9. Additional experiments for depth completion. Quantitative results of depth estimation on KITTI validation dataset. (Unit: meter, **Bold**: Best)

	KITTI Depth Completion			
	RMSE	MAE	iRMSE	iMAE
CSPN	0.8229	0.2264	0.0026	0.0011
Naïve	0.8489	0.2389	0.0025	0.0011
Ours	**0.8182**	**0.2190**	**0.0024**	**0.0010**

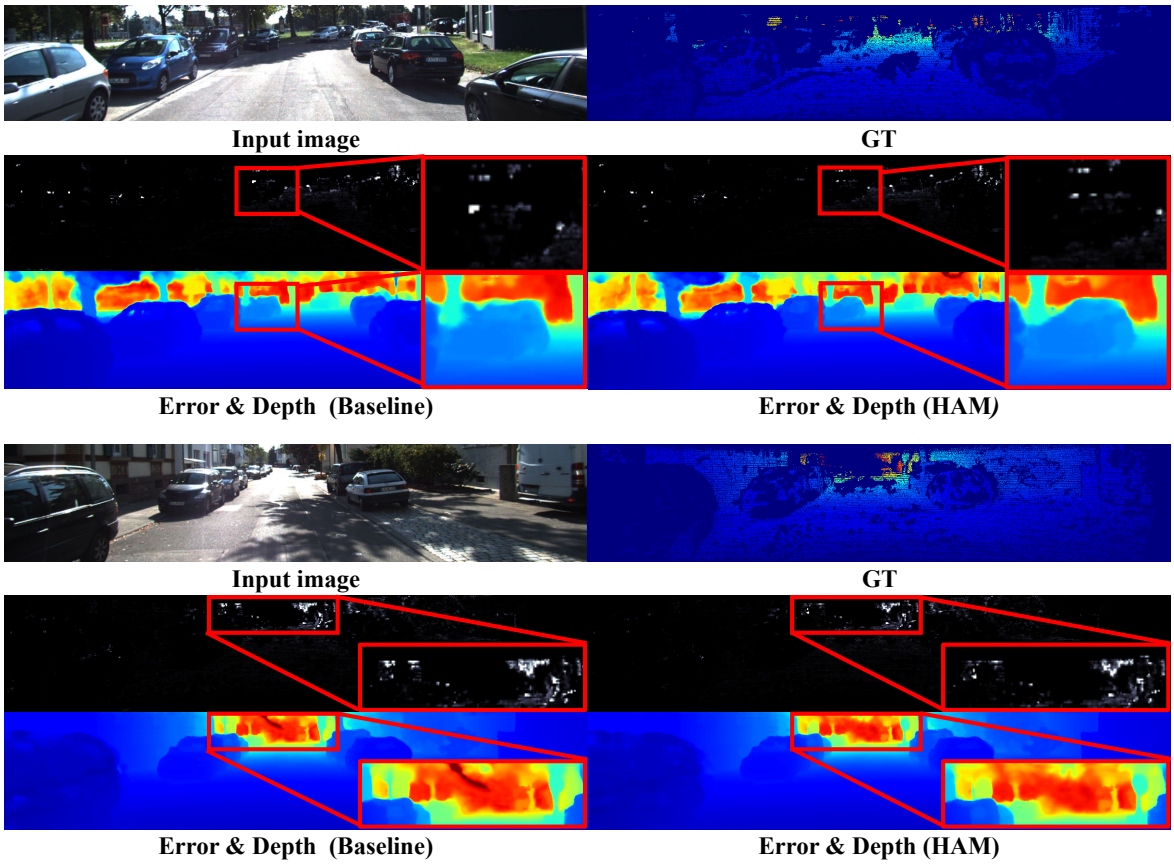


Figure 12. Qualitative results of KITTI-DC dataset (Uhrig et al., 2017).

E. More experiments for Depth Completion

E.1. Additional Experiment on Larger Dataset

To demonstrate the robustness of our method, we conduct additional experiments on outdoor dataset, the KITTI Depth Completion dataset (Uhrig et al., 2017). The KITTI-DC dataset consists of over 90K RGB and LiDAR pairs. As follows (Park et al., 2020), we exclude top 100 rows in images where are a region without LiDAR projection by cropping 240 (height) $\times$ 1216 (width) patches for both training and test.

Our HAM is trained for 30 epochs with both L_1 and L_2 losses, and the initial learning rate is decayed by 0.5 at every 5 epochs after the first 10 epochs. In the training phase, we choose a batch size of 12. In the same experimental setup used in the manuscript, we set the size of the kernel γ to 3 and utilize a 2-dimensional hyperbolic convolution operation. We also conduct a variety of curvatures, *i.e.,* $\kappa \in \{0.1, 0.5, 1.0, 1.5, 2.0\}$, and obtain the best performance when $\kappa = 1.5$.

Due to the sparsity of the depth map in the KITTI-DC dataset (Geiger et al., 2013), we need to rearrange pixels on higher curvature hyperbolic space than that of NYUv2 dataset. We observe that the Poincaré ball with the high curvature κ yields a low variance affinity map because the geodesic distance distribution becomes smoother as the curvature κ increases. By

smoothing the affinity among pixels, propagating initial depth points between two pixels which are located far away and have big difference values works well.

As shown in Tab. 9-(a), our method outperforms the baseline with the fully convolutional layer (Cheng et al., 2018) as well as the basic hyperbolic convolution layers (Ryohei et al., 2021). In particular, the qualitative results from our HAM in Fig. 12 yield low error when estimating thin structures and objects' boundaries.

E.2. Various Spatial Propagation Schemes: Further Details of Figure 8-(a) in the manuscript

We provide further details of the experiments on various propagation schemes. We show the quantitative and qualitative results of applying HAM to convolutional propagation, non-local propagation, and dynamic (attention-based) propagation.

Convolution-based spatial propagation with HAM. Firstly, HAM with convolutional propagation method (CSPN) shows significant improvement compared to baseline models as shown in Tab. 10. The gap becomes larger in rare environments, *e.g.,* the difference between baseline and our HAM w.r.t. RMSE is 0.014m for 500 sample points and 0.067m for 100 sample points, respectively. Fig. 14 demonstrates CSPN with HAM is robust to propagation for thin structure, which indicates it can construct high-fidelity affinities than baseline models.

Non-local spatial propagation with HAM. We conduct experiments with NLSPN (Park et al., 2020) that proposes a non-local propagation with deformable convolution layers. Similar to our experiments in the main paper, we position our HAM at the end of the affinity branch, which induces faithful spatial propagation. As shown in Tab. 11, our method shows comparable performance in dense input setup, and shows significant improvement with respect to the sparse initial seeds (*i.e.,* 250 and 100). The qualitative results (Fig. 15) demonstrates that HAM preserves thin structures and homogeneous surfaces, compared to other methods, which suffer from preserving connectivity in harsh condition. This experiment shows that our HAM applies to affinity maps with non-local connectivity from deformable convolutions, and the HAM synergies well with the non-local propagation approach.

Attention-based spatial propagation with HAM. We also provide quantitative results and examples of depth completion tasks on DYSPN (Lin et al., 2022), which proposes an attention-based dynamic approach to fixed affinities and achieves SoTA performance in the depth completion task. As shown in Tab. 12 below, the hyperbolic method demonstrates a significant improvement in the sparse setup (sample 100 or 250), except dense sample setup. We argue that since the DYSPN utilizes spatial and sequential attention to adjust the dynamic affinity map in the 2D image domain (i.e., Euclidean space), the replacement of the affinity layer with our HAM does not show additional performance gain. It seems that we can adopt dynamic hyperbolic representation (*e.g.,* inference curvature for each iteration like (Gao et al., 2021)) for the spatial propagation scheme. Note that the hyperbolic representation shows consistent improvement, especially in harsh conditions (*i.e.,* sparse environment of prior information). Fig. 16 indicates that HAM is robust to homogeneous surfaces, compared to other methods that suffer from capturing depth boundaries in the sparse environment. This experiment shows that our HAM is applicable for non-linear propagation scheme and the advantage of the hyperbolic layer is enlarged when our HAM is adopted, compared to the naïve approach.

Table 10. Depth completion results w.r.t. the number of seed points. Note that the baseline is Convolutional Spatial Propagation Network (CSPN (Cheng et al., 2018)), which adopts a recursive convolution operation.

	Convolutional						
Sample #	Methods	RMSE	MAE	iRMSE	iMAE	REL	$\delta^1_{1.25}$
500	CSPN	0.116	0.048	0.018	0.007	0.017	0.993
500	Naïve	0.108	0.043	0.017	**0.006**	0.015	**0.994**
500	Ours	**0.102**	**0.040**	**0.016**	**0.006**	**0.014**	**0.994**
250	CSPN	0.149	0.066	0.024	0.010	0.023	0.987
250	Naïve	0.147	0.064	0.023	0.010	0.023	0.989
250	Ours	**0.136**	**0.057**	**0.021**	**0.009**	**0.020**	**0.990**
100	CSPN	0.366	0.203	0.056	0.030	0.068	0.918
100	Naïve	0.319	0.161	0.050	0.025	0.056	0.949
100	Ours	**0.299**	**0.146**	**0.048**	**0.023**	**0.051**	**0.953**

Table 11. Depth completion results w.r.t. the number of seed points. Note that the baseline is Non-Local Spatial Propagation Network (NLSPN (Park et al., 2020)) by predicting the offset for each pixel to determine where the information should come from.

	Non-Local						
Sample #	Methods	RMSE	MAE	iRMSE	iMAE	REL	$\delta^1_{1.25}$
500	NLSPN	**0.094**	**0.037**	**0.014**	**0.005**	**0.013**	**0.995**
500	Naïve	0.099	0.042	0.016	0.007	0.015	**0.995**
500	Ours	0.097	0.041	0.015	0.006	0.014	**0.995**
250	NLSPN	0.129	0.057	0.020	0.009	0.020	0.991
250	Naïve	0.126	0.055	0.020	0.009	0.019	0.991
250	Ours	**0.124**	**0.051**	**0.019**	**0.008**	**0.018**	**0.992**
100	NLSPN	0.327	0.171	0.051	0.026	0.056	0.934
100	Naïve	0.291	0.161	0.047	0.025	0.055	0.951
100	Ours	**0.258**	**0.134**	**0.039**	**0.020**	**0.045**	**0.965**

Table 12. Depth completion results w.r.t. the number of seed points. Note that the baseline is Dynamic Spatial Propagation Network (DYSPN (Lin et al., 2022)), which designs a non-linear propagation model with spatial-sequential attention.

	Dynamic						
Sample #	Methods	RMSE	MAE	iRMSE	iMAE	REL	$\delta^1_{1.25}$
500	DYSPN	**0.092**	**0.037**	**0.014**	**0.005**	**0.012**	**0.996**
500	Naïve	0.096	0.039	**0.014**	0.006	0.013	**0.995**
500	Ours	0.093	**0.037**	**0.014**	**0.005**	0.013	**0.995**
250	DYSPN	0.127	0.056	**0.019**	**0.008**	0.019	**0.992**
250	Naïve	0.124	**0.052**	**0.019**	**0.008**	**0.018**	**0.992**
250	Ours	**0.123**	0.053	**0.019**	**0.008**	**0.018**	**0.992**
100	DYSPN	0.322	0.171	0.054	0.027	0.059	0.938
100	Naïve	0.307	0.167	0.049	0.025	0.056	0.946
100	Ours	**0.269**	**0.143**	**0.040**	**0.021**	**0.047**	**0.963**

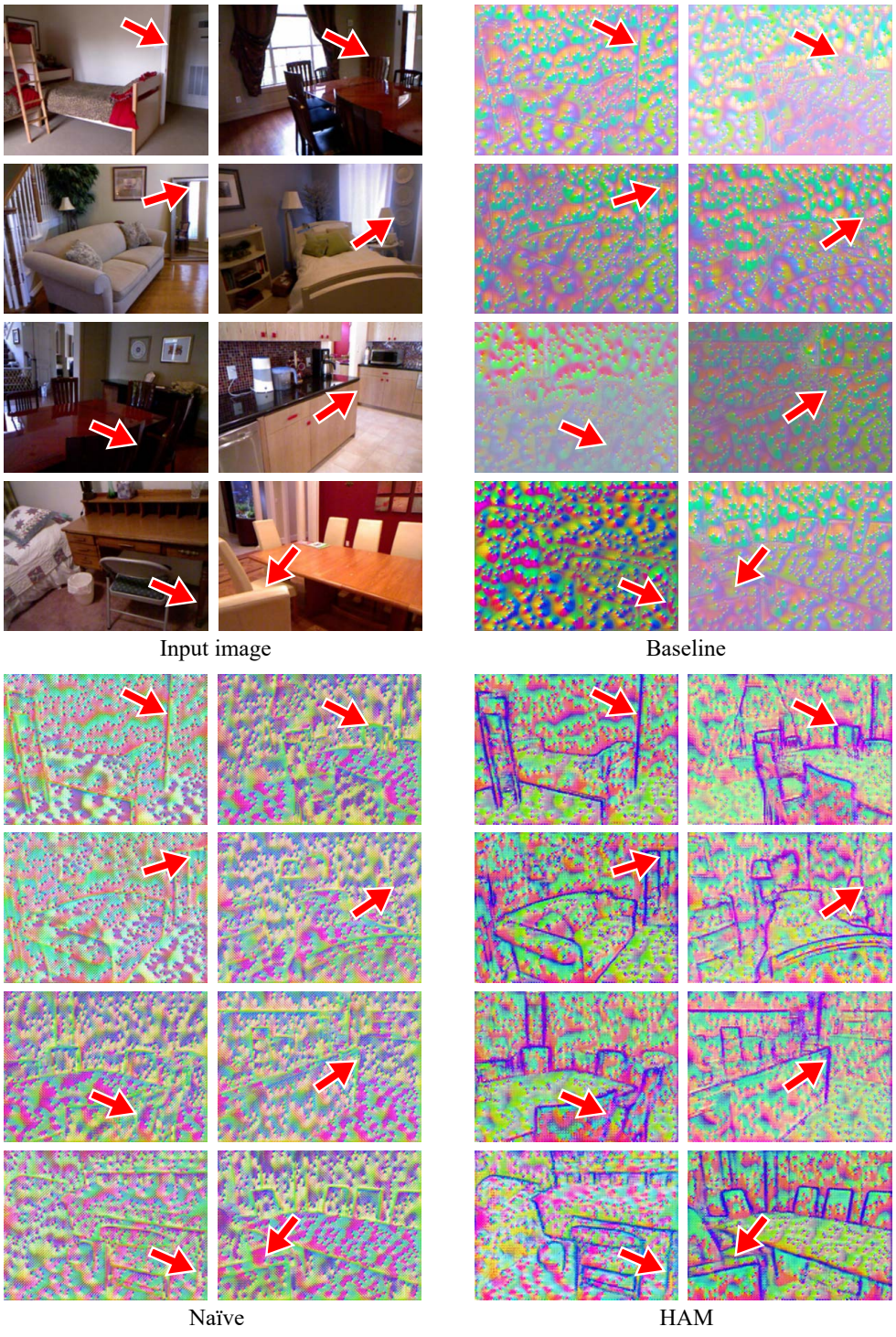


Figure 13. An illustration of affinity maps from our baseline (Cheng et al., 2018), naïve and HAM in NYUv2 dataset. It shows that affinity maps from our HAM well preserve edge information (*i.e.,* boundary) where smooth intensity changes occur. Note that red arrows indicate object boundaries where noises or smooth intensity changes occur in images.

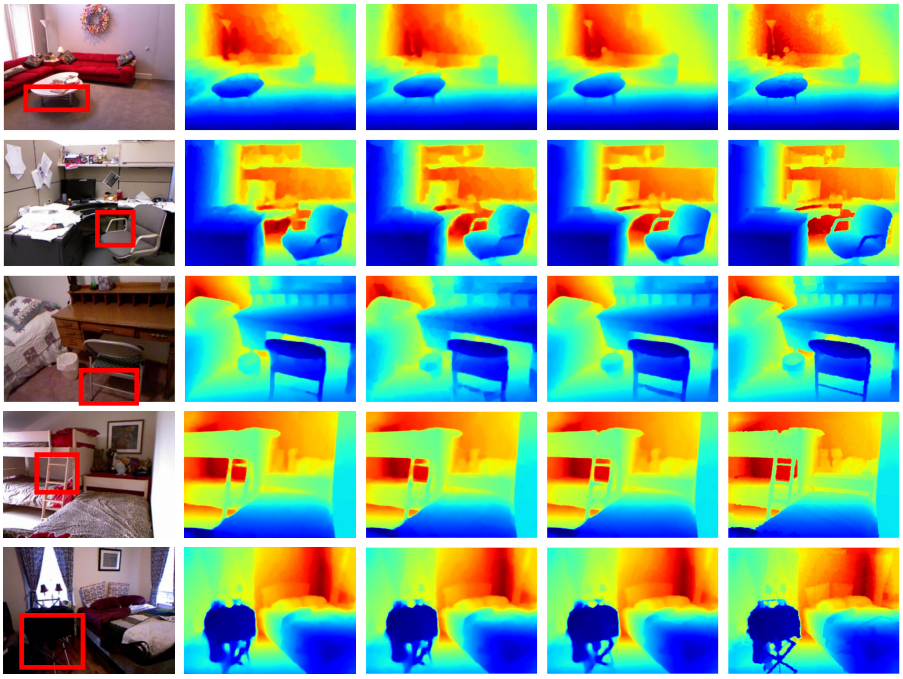


Figure 14. Qualitative comparison on NYUv2 (Silberman et al., 2012). We compare the baseline model (CSPN (Cheng et al., 2018)) and naïve with our HAM. Note that we visualize the case of 500 depth samples in Tab. 10. Our HAM particularly demonstrates the strength for thin structures (*e.g.,* red-colored rectangles).

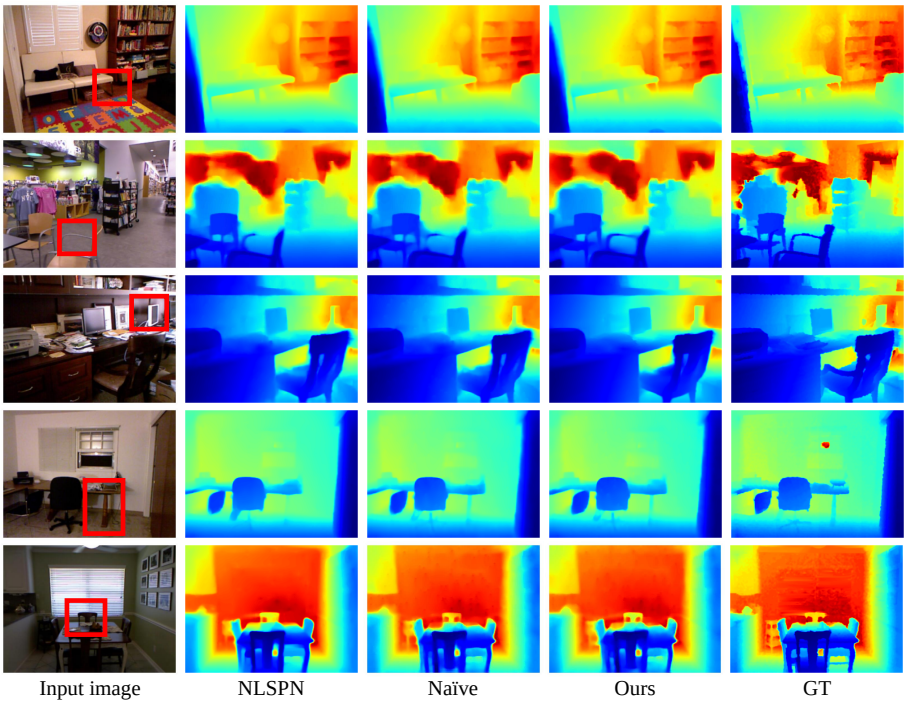


Figure 15. Qualitative comparison on NYUv2 (Silberman et al., 2012). We compare the baseline model (NLSPN (Park et al., 2020)) and naïve with our HAM. Note that we visualize the case of 250 depth samples in Tab. 11. Our HAM particularly demonstrates the strength for thin structures (*e.g.,* red-colored rectangles).

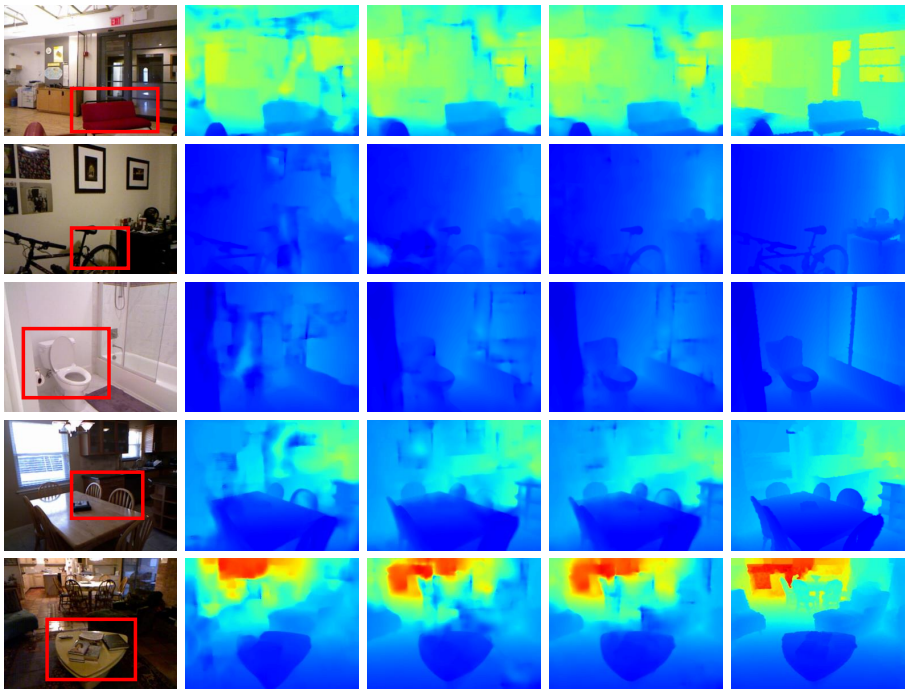


Figure 16. Qualitative comparison on NYUv2 (Silberman et al., 2012). We compare the baseline model (DYSPN (Lin et al., 2022)) and naïve with our HAM. Note that we visualize the case of 100 depth samples in Tab. 11. Our HAM particularly demonstrates the strength for homogeneous surface and slanted objects (*e.g.,* red-colored rectangles).

F. More Experiments for Semantic Segmentation

F.1. Additional Experiment on larger dataset

	Semantic Segmentation (Unit: mIoU, Bold: Best)												
	cat	parking meter	sheep	frisbee	sports ball	surfboard	tennis racket	cup	sandwich	orange	donut	potted plant	curtain
Sim-Deeplab	70.01	54.18	44.45	33.97	19.11	34.22	11.98	22.92	54.45	25.39	25.97	21.93	34.18
DifNet	74.17	39.75	43.56	34.19	0.56	29.17	13.85	20.19	55.64	26.58	31.97	29.06	33.35
Naïve	73.64	69.56	45.73	32.80	12.86	39.25	11.84	23.32	55.18	24.76	41.24	29.36	34.38
Ours	75.08	76.77	55.97	56.81	51.57	42.13	14.66	24.86	56.32	35.03	46.22	32.35	35.61
	desk stuff	fruit	hill	playing field	railroad	rug	skyscraper	tent	towel	wall concrete	wall stone	car	bird
Sim-Deeplab	24.70	20.65	13.92	55.85	44.80	6.30	24.05	3.64	4.93	3.57	10.08	66.98	44.91
DifNet	25.59	15.35	13.59	59.67	47.22	6.24	28.44	8.58	4.28	1.14	12.16	69.01	48.31
Naïve	30.60	21.07	21.17	61.70	50.38	9.54	29.04	9.79	3.85	3.02	12.80	73.78	52.88
Ours	31.71	24.17	22.40	66.52	54.77	14.37	31.85	13.91	7.19	10.51	13.21	73.72	52.68
	horse	elephant	wine glass	pizza	cake	bed	laptop	fence	pavement	plastic	road	mIoU	Pixel Accuracy
Sim-Deeplab	56.39	77.19	58.99	70.77	23.84	28.39	47.05	29.43	25.95	0.07	46.08	23.93	60.84
DifNet	56.73	79.70	56.57	64.75	35.32	37.67	49.21	32.52	30.72	0.20	49.80	24.46	**63.10**
Naïve	62.25	79.77	62.45	70.04	27.49	35.55	57.97	30.61	33.66	7.58	44.76	24.76	62.39
Ours	61.92	79.04	60.95	68.58	33.44	35.66	54.93	32.14	32.87	7.26	49.24	**25.22**	62.27

Table 13. **Quantitative results of semantic segmentation on COCO-Stuff10K.**

We provide more details of semantic segmentation results on the COCO-Stuff10K (Caesar et al., 2018), which consists of 10,000 images from 171 classes, and is split into 9,000 images in the training set and 1,000 images in the test set. We note that ResNet18 is utilized as the backbone to train our method and the comparison methods for 200 epochs. As reported in Tab. 13, our HAM outperforms the comparison methods. We observe that our method obtains performance gains for thin-structure and sharp boundaries of objects like surfboard, tent, cup, and fruit in the dataset. Compared to other experiments in the main manuscript, the COCO-Stuff10K dataset contains a variety of classes in images, which requires sophisticated connectivity among pixels. To handle this issue, it is necessary to selectively encode important features like our HAM.

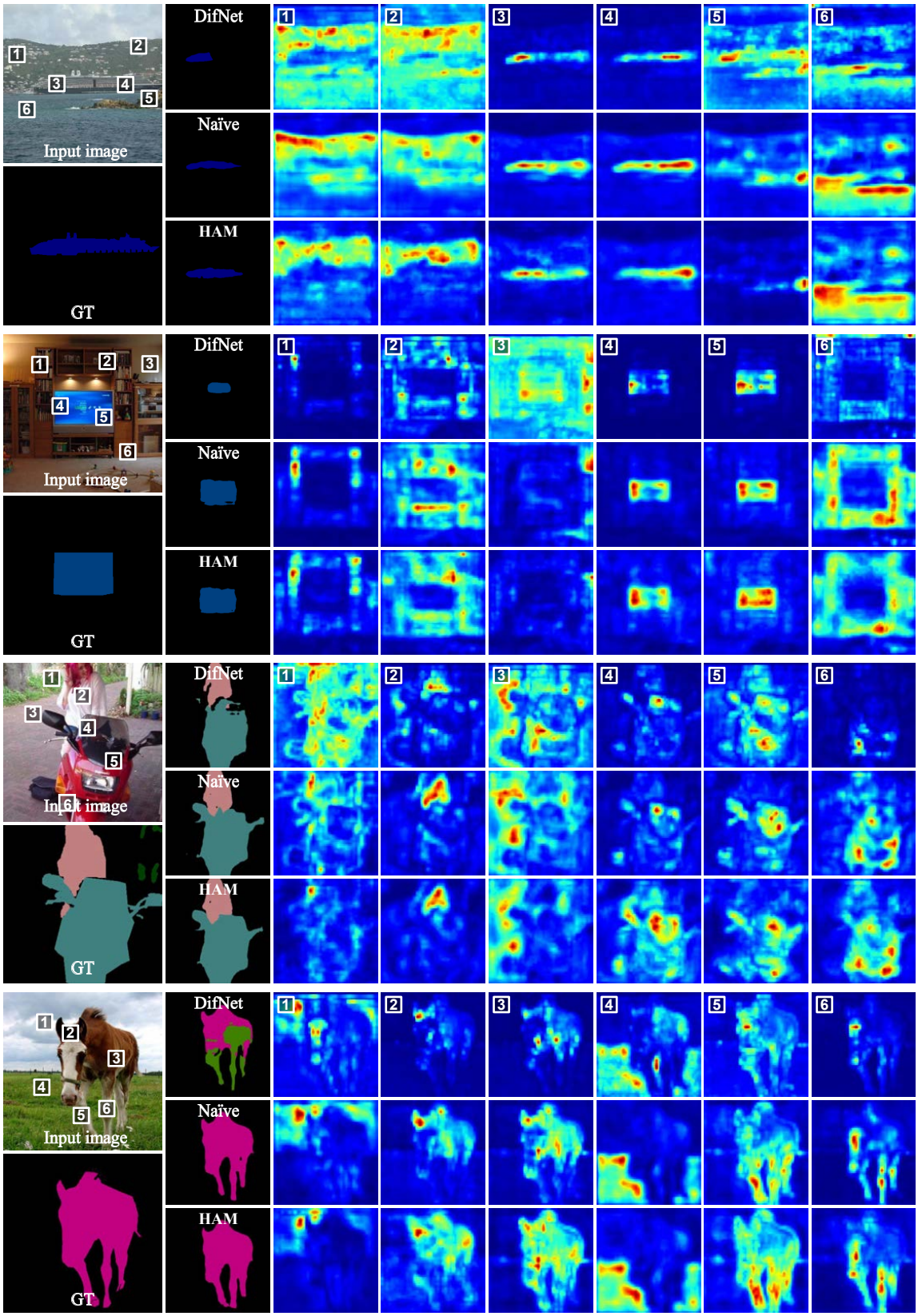


Figure 17. Visualizations of similarity maps on Pascal VOC 2012 dataset (Everingham et al., 2015). We visualize similarity maps under six selected nodes for each input image. Note that red colors on similarity maps represent highly similar nodes with respect to each selected node.

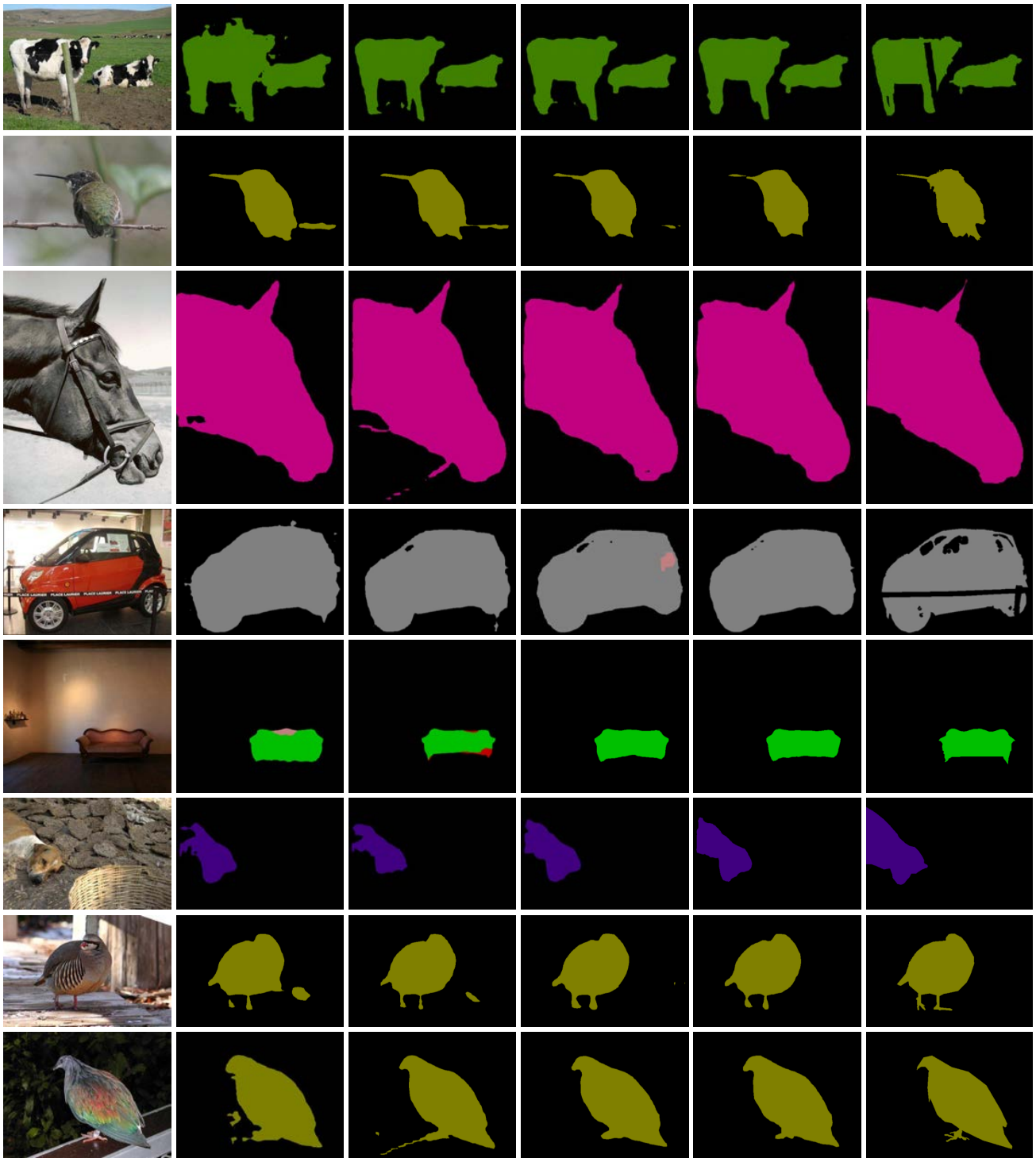


Figure 18. Examples of semantic segmentation results on Pascal VOC 2012 dataset (Everingham et al., 2015).

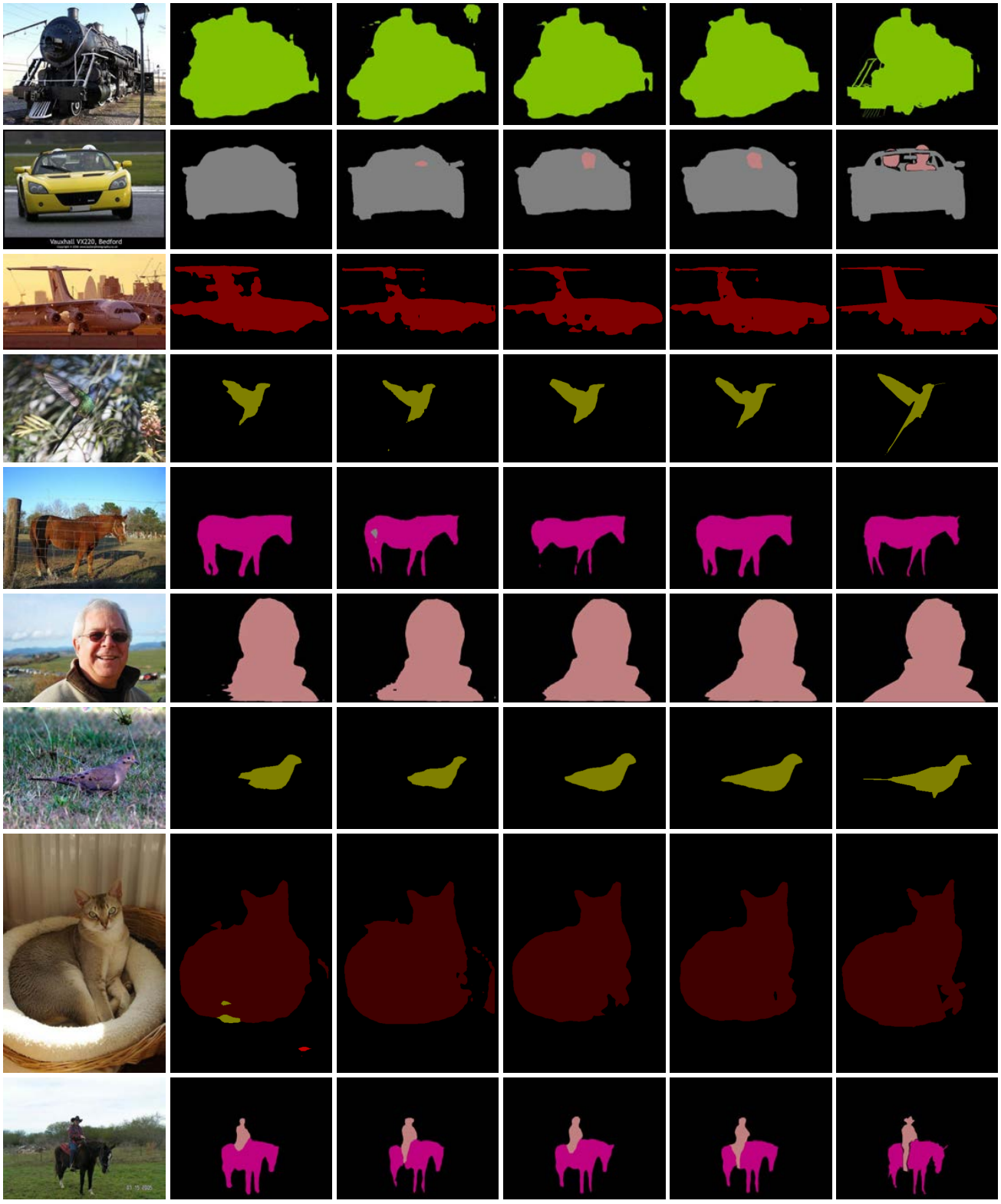


Figure 19. More examples of semantic segmentation results on Pascal VOC 2012 dataset (Everingham et al., 2015).

Figure 20. More examples of semantic segmentation results on Pascal VOC 2012 dataset (Everingham et al., 2015).