

FAHNet: Accurate and Robust Normal Estimation for Point Clouds via Frequency-Aware Hierarchical Geometry

Chengwei Wang¹, Wenming Wu^{1,2}, Yue Fei¹, Gaofeng Zhang³, Liping Zheng^{†1}

¹School of Computer Science and Information Engineering, Hefei University of Technology

²Anhui Province Key Laboratory of Industry Safety and Emergency Technology (Hefei University of Technology), Hefei 230601, Anhui, P.R. China

³School of Software, Hefei University of Technology

Abstract

Point cloud normal estimation underpins many 3D vision and graphics applications. Precise normal estimation in regions of sharp curvature and high-frequency variation remains a major bottleneck; existing learning-based methods still struggle to isolate fine geometry details under noise and uneven sampling. We present FAHNet, a novel frequency-aware hierarchical network that precisely tackles those challenges. Our Frequency-Aware Hierarchical Geometry (FAHG) feature extraction module selectively amplifies and merges cross-scale cues, ensuring that both fine-grained local features and sharp structures are faithfully represented. Crucially, a dedicated Frequency-Aware geometry enhancement (FA) branch intensifies sensitivity to abrupt normal transitions and sharp features, preventing the common over-smoothing limitation. Extensive experiments on synthetic benchmarks (PCPNet, FamousShape) and real-world scans (SceneNN) demonstrate that FAHNet outperforms state-of-the-art approaches in normal estimation accuracy. Ablation studies further quantify the contribution of each component, and downstream surface reconstruction results validate the practical impact of our design.

CCS Concepts

• *Computing methodologies* → *Point-based models; Parametric curve and surface models;*

1. Introduction

With the rapid advancement of autonomous driving [YCSL24] and virtual reality [GRAS*21], 3D point cloud data has emerged as a vital resource for accurately representing real-world objects and complex environments. Among the core tasks in point cloud analysis, normal estimation plays an essential role by inferring local surface orientations from raw point samples. Accurate normals are crucial for many downstream applications, such as surface reconstruction [KBH06], [KH13a], graphic rendering [WWR22], semantic segmentation [CO18], classification [WQF19], target tracking [GZXS24], where subtle geometric details often provide significant cues.

Traditionally, normal estimation has relied on local geometric primitive fitting—using techniques such as principal component analysis (PCA) to obtain local planes or polynomial surface fitting to determine the surface orientation [HDD*92] [MN03] [HLZ*09]. Although these methods are straightforward and computationally efficient, they are highly sensitive to noise, density variations, and require careful parameter tuning. Consequently, their performance often deteriorates when handling complex geometries or varying noise levels.

In contrast, recent years have witnessed a surge in deep learning-based methods that address these limitations by leveraging the representational power of neural networks. Broadly, these approaches fall into two categories. The first type, like IterNet [LOM19], DeepFit [BSG20], AdaFit [ZLD*21], GraphFit [LZW*22], adopts a fitting-based paradigm where deep networks are used to predict point-wise weights that guide weighted least squares fitting. The second employs regression-based strategies that directly predict the normal vector from features extracted via multilayer perceptrons (MLPs). PCPNet [GKOM18] was the first to employ neural networks to regress normals from unstructured point clouds directly. Zhang et al. [ZCZ*22a] transfer the point-wise weights from fitting-based methods into a regression-based framework, enabling the network to learn under the guidance of latent geometric features. Subsequent studies [ZCZ*22b], [LLC*22], [LFS*23], [XLW*23], [WZL*24] have consistently elevated normal estimation accuracy to unprecedented levels.

Despite significant progress, critical challenges remain. First, to extract accurate local geometric features, many methods employ a KNN-based graph features learning strategy. However, the choice of KNN scale size significantly impacts model performance; a large scale may lose fine-grained local details, while a small one heightens sensitivity to noise. Although recent works adopt multi-scale strategies to extract local geometric features for both accuracy and

[†] Corresponding Author

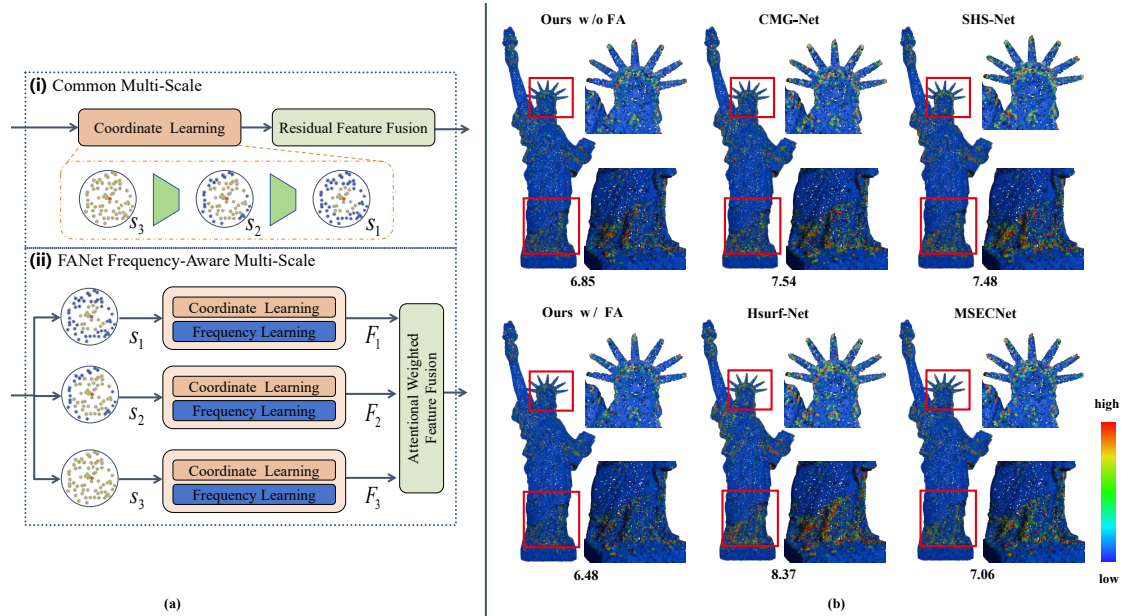


Figure 1: Comparing our FAHNet with common multi-scale normal regression methods. (a) Previous regression methods typically employ downsampling and dimensional transformations with MLPs to achieve multi-scale feature fusion, and learning only in the coordinate space. In contrast, our method employs a parallel hierarchical architecture, combined with attention mechanisms for fusing hierarchical multi-scale features. It extends the learning paradigm by integrating frequency-domain representations of point clouds with coordinate space features through our Frequency-Aware geometry enhancement (FA) module. (b) Visualization of normal RMSE errors, our method outperforms the comparison methods, and the FA module further enhances estimation accuracy for complex structures and sharp features.

robustness [GKOM18], [XLW*23], [WZL*24]. Those fused representations often degrade through subsequent downsampling and dimension transformations, preventing the network from capturing fine-grained surface structures. Second, recovering accurate normals around sharp edges and high-curvature areas remains challenging, and this difficulty is further exacerbated by noise and uneven sampling. Moreover, Rahaman et al. [RBA*19] point out that MLPs tend to learn low-frequency components of the data, which hinders effective analysis of high-frequency variation and sharp features in normal estimation tasks. These shortcomings in feature extraction severely constrain overall normal-estimation performance.

To address the issues mentioned above, we propose FAHNet. This novel end-to-end architecture combines hierarchical multi-scale spatial structures with frequency awareness for accurate and robust normal estimation. As illustrated in **Fig. 1**, FAHNet's backbone is a Frequency-Aware Hierarchical Geometry (FAHG) module. FAHG employs multiple resolution streams that concurrently encode both contextual structures and fine-grained details across distinct spatial hierarchies. This design preserves geometric cues at each hierarchical layer, thereby strengthening the network's capacity for multi-scale feature analysis. Unlike prior approaches that aggregate hierarchical features via successive downsampling and dimension transformations, we introduce an Attentional Hierarchical Feature Fusion (AHFF) module that dynamically selects keypoint features and combines cross-scale representations, markedly enhancing the network's capacity to learn fine-grained surface details. Moreover, to counteract the inherent low-frequency bias of MLPs,

we embed a Frequency-Aware geometry enhancement (FA) branch at each hierarchical layer. FA enhances the network's representation of high-frequency geometry information, resulting in more accurate estimations in regions with complex structures and sharp features, particularly under noisy conditions.

We evaluate our approach through comprehensive experiments on synthetic datasets PCPNet [GKOM18] and FamousShape [LFS*23], which encompass scenarios with non-uniform density, discontinuous point clouds, noise, thin and sharp structures, as well as on real-world datasets SceneNN [HPN*16] to demonstrate its generalization capability. In direct comparisons with recent state-of-the-art methods, FAHNet achieves the highest prediction accuracy with the fewest parameters. Ablation studies further confirm the effectiveness of each module. Our major contributions can be described as follows:

- We propose FAHNet, a novel end-to-end normal estimation approach featuring a hierarchical multi-scale backbone and an attentional fusion module, enabling precise and fine-grained geometry feature extraction.
- We introduce a frequency-aware geometry enhancement branch that captures high-frequency details in point clouds, substantially enhancing the model's ability to recover intricate surface structures under variable density and noise.
- Extensive experiments on multiple diverse datasets demonstrate that our method outperforms recent state-of-the-art approaches in both prediction accuracy and robustness.

2. Related Work

2.1. Traditional Methods

Classic normal estimation techniques typically derive the surface normal at a point by analyzing its local neighborhood. The most prevalent approach is based on Principal Component Analysis (PCA), HOPPE et al. [HDD*92] compute the covariance matrix of points within a fixed-size patch and defines the normal as the eigenvector corresponding to the smallest eigenvalue. Variants [MN03] [HLZ*09] have emerged to improve robustness in noisy settings—for example, Moving Least Squares (MLS) [Lev98] and truncated Taylor expansion fitting (n-jet) [CP05] incorporate higher-order fittings over larger neighborhoods to mitigate noise influence. Other methods, such as local spherical surface fitting [GG07] and [ASL*17], further attempt to account for variations in patch scale.

To preserve finer geometric details, alternative techniques such as Voronoi diagrams [AB98] [ACSTD07] [MOG11], Hough transform [BM12], and plane voting [ZCL*19] have been introduced. Despite offering more accurate results, these methods require meticulous manual parameter tuning (e.g., neighborhood size or voting thresholds). More recently, iPSR [HWW*22] uses iterative Screened-PoissonSurfaceRecon struction (SPSR) [KH13b] alternately infers normals from the current mesh and refines the surface. Marin et al. [MOW24] used a parameter-free spheres-of-influence graph (SIG) to capture the connectivity of the point cloud and then estimate normals. However, their method did not consider large-scale point clouds and remains sensitive to data corruption. SNO [HFZ*24] proposes a stochastic L-BFGS optimization of a signed-uncertainty field for normal orientation. WNNC [LSL24] utilizes the normal direction to optimize the winding number formula for globally consistent normals. In summary, traditional methods offer interpretability and strong theoretical support but often struggle to capture sharp features and detailed local information in diverse and noisy datasets. Some also require multi-stage optimization, which leads to poor time efficiency and difficulty handling large-scale point clouds.

2.2. Learning-Based Methods

The past years have witnessed a surge in deep learning-based normal estimation techniques, largely driven by the powerful feature extraction capabilities of neural networks. These methods can be broadly categorized into two groups: deep surface fitting approaches and regression-based approaches.

Surface Fitting. Surface Fitting Approaches learn to predict point-wise weights that guide a weighted surface polynomial fitting on local patches. Early works like DeepFit [BSG20] and Iter-Net [LOM19], leverage these learned weights to perform a more robust surface fit. Subsequent advances, such as incorporating multi-scale feature aggregation in AdaFit [ZLD*21] or integrating graph convolutional layers, as seen in GraphFit [LZW*22], further refine the weight prediction process. Zhou et al. [ZJW*21] improve normal estimation by simplifying the surface fitting process through a Top-K selection strategy and a refined point update mechanism, thereby enhancing the robustness of the method. Zhang et al. [ZCZ*22a] propose a geometry-guided deep learning framework that explicitly integrates geometric constraints into surface normal estimation. Du

et al. [DYW*23] reexamine the approximation error in 3D surface fitting for normal estimation, offering critical insights that pave the way for designing more accurate techniques. Li et al. [LLZ*24] introduces a geometry-aware hierarchical learning framework that integrates the geometry-aware hierarchical graph representation into the fitting process. These methods excel at reducing the dependence on manual parameter tuning and often demonstrate improved performance in clean conditions. However, they remain constrained by the fixed order of the polynomial fitting, making them susceptible to overfitting or underfitting whenever the underlying surface complexity varies significantly.

Regression-Based. Regression-Based methods treat normal estimation as a direct regression problem and leverage network architectures to infer normal vectors from raw point cloud patches. Early methods, such as HoughCNN [BM16], convert point cloud data into 2D representations (e.g., Hough space or height maps), which are then processed by convolutional neural networks (CNNs). Although this tactic benefits from mature 2D network architectures, it inevitably loses some intrinsic 3D geometric details. To counteract this, subsequent approaches exemplified by PCPNet [GKOM18] operated directly on unstructured 3D point clouds using PointNet [QSMG16] as the backbone for multi-scale feature extraction. This is the first method to implement direct normal regression from local point patches. Further innovations have introduced additional cues through point-voxel architectures, local plane constraints, and the incorporation of supporting information such as initial normals and height maps. Hashimoto et al. [HS19] incorporates spatial structural information into point cloud models for normal estimation with a dual-stream network. Ben et al. [BSLF19] learn a multi-scale feature representation that automatically identifies the optimal local patch neighborhood scale. Zhou et al. [ZHLL19] propose an innovative normal estimation approach that integrates local plane constraints with a multi-scale selection mechanism. Refine-Net [ZCZ*22b] introduces a dedicated refinement network that iteratively enhances initial normal estimates on noisy point clouds through multi-scale feature fusion and adaptive correction strategies. HSurf-Net [LLC*22] and SHS-Net [LFS*23] ingeniously transform the regression task on point clouds into hyper surface representations prediction, achieving robust and accurate normal estimation. NeAF [LZM*23] learns neural angle fields for point normal estimation, enabling the robust capture of subtle angular variations in point clouds for more precise surface geometry representation. MSECNet [XLW*23] enhances normal estimation by integrating edge detection technology into a multi-scale edge conditioning stream framework, thereby achieving significantly higher accuracy. CMG-Net [WZL*24] introduces a novel Chamfer Normal Distance metric to resolve normal direction inconsistencies caused by noise in point clouds, thereby significantly enhancing the robustness of the model's predictions under noisy conditions. Li et al. [LLX*24] introduces a transformer-based framework that fuses visual semantic cues to extract refined surface geometry. While regression-based methods demonstrate promising results on well-structured, clean point clouds, they still rely heavily on network architecture designs and often struggle to capture subtle geometric variations under noisy conditions.

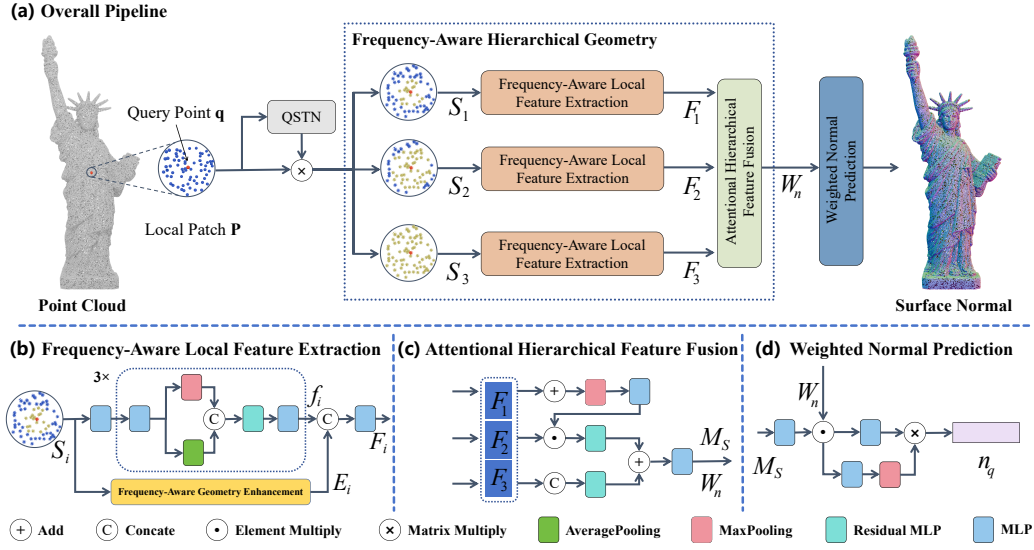


Figure 2: Architecture of the proposed method. (a) Overall Pipeline of FAHNet, our method predicts normals in an end-to-end manner. (b) Frequency-Aware Local Feature Extraction, composed of a coordinate branch and a frequency branch. (c) Attentional Hierarchical Feature Fusion. (d) Weighted Normal Prediction.

3. Method

3.1. Challenges

Despite significant progress, learning-based methods still confront the following challenges. The primary difficulty lies in achieving **accurate and robust feature extraction**. The inherent lack of connectivity among points, coupled with noise, makes it difficult to extract fine-grained geometric details. This challenge is exacerbated with complex surface variations, where preserving sharp edges requires robust and adaptive feature extraction strategies. Furthermore, limitations in existing frameworks (e.g., MLPs, classical graph convolutions) also restrict the network's ability to extract accurate geometric details directly from the point cloud. The second is **Critical detail preservation in multi-scale fusion**. Scale size selection is critical. While larger scales tend to provide robustness against noise, they often oversmooth essential shape details. Conversely, smaller scales capture finer geometric nuances but exhibit higher sensitivity to noise. Moreover, integrating features across these scales introduces additional complexities that can suppress critical fine-grained information.

3.2. Overview

To address the challenges above, we propose FAHNet, a novel end-to-end approach designed for accurate and robust point cloud normal estimation. The overall architecture is shown in **Fig. 2(a)**, for any given point q in $X = \{x_i \in \mathbb{R}^3\}_{i=1}^T$, T denotes the total number of points in the point cloud, we use KNN algorithm to construct its corresponding local patch $P = \{x_i \in \mathbb{R}^3\}_{i=1}^N$, N is the number of neighbors. To mitigate training instability caused by variations in point cloud orientation, we first apply normalization and a rotation transformation called Quaternion Spatial Transformer Network (QSTN), which is similar to [GKOM18]. Subsequently, we

feed the constructed hierarchical sub-patches S_i (the points highlighted in yellow), along with their corresponding graph features, into the Frequency-Aware Hierarchical Geometry (FAHG) feature extraction module (Sec. 3.4) to obtain fine-grained geometric features separately. These features are then fused via an Attentional Hierarchical Feature Fusion (AHFF) module (Sec. 3.4), specifically designed to capture and aggregate discriminative information across different hierarchical scales. To further enhance the network's ability to capture sharp geometric details, we introduce the Frequency-Aware geometry enhancement (FA) module (Sec. 3.3), which extracts high-frequency variations directly from the raw point cloud and seamlessly integrates them into the feature extraction pipeline. Finally, the network outputs the predicted normal vector $n_q \in \mathbb{R}^3$ for query point q .

3.3. Frequency-Aware Local Feature Extraction

As shown in the **Fig. 2(b)**, within the Frequency-Aware Local Feature Extraction (FALFE) module, each normalized sub-patch S_i and the initial KNN graph feature are passed through the coordinate branch to extract distance features and through the frequency branch to extract spectral features, and then the two outputs are merged via residual MLPs.

In the coordinate learning branch, the geometric processed feature f_i can be expressed by the following equations:

$$f_{i+1} = \Phi_3(\odot(\Phi_2((\text{MEAN} \odot \text{MAX})\{\Phi_1(f_i \cdot w_i)\}))) \quad (1)$$

where $i = 1, 2, \dots, S$, and S total hierarchy level numbers and i is the the current index, $\Phi_1, \Phi_2, \Phi_3, \Phi_4$ is MLP layers, \odot is the concatenation, \odot is the residual MLP for skip-connection, and $(\text{MEAN} \odot \text{MAX})\{\cdot\}$ represents concatenated average pooling and maxpooling features over current patch S_i . w_i is a linear distance

graph node weight, and for each node $x_i \in S_i$, its weight is computed by

$$w_{x_i} = \frac{\beta_{x_i}}{\sum_{i=1}^{N_i} \beta_{x_i}}, \beta_{x_i} = \text{sigmoid}(\alpha_1 - \alpha_2 \|x_i - q\|_2), \quad (2)$$

where N_i is the point number of S_i , α_1 and α_2 are learnable parameters, and w_{p_s} enables the network to tune the weights of patch points in a distance-aware manner. Besides, we stack the coordinate branch three times.

In recent years, several studies [RBA*19], [MST*20], [TSM*20] have pointed out that MLPs tend to learn the low-frequency features present in data preferentially, exhibiting a poorer ability to fit high-frequency functions. This phenomenon is also evident in normal estimation tasks, where MLPs yield significantly lower performance in estimating complex regions with high-frequency variation compared to those with low-frequency variation, resulting in an over-smoothing pitfall. The vertex in the complex area and its nearby neighbors often have nearly identical coordinate graph features; however, their ground-truth normals may differ by more than 30 degrees. MLP struggles to predict sharp normal changes from these similar coordinate features.

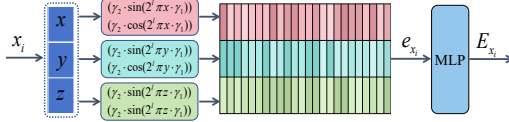


Figure 3: Illustration of per point cloud mapping process in FA.

This suggests that a purely coordinate architecture is ineffective in capturing drastic curvature changes of the underlying surface from point clouds. To address this issue, as shown in the Fig. 3, we have designed a Frequency-Aware Geometry Enhancement (FA) branch that extracts high-frequency point cloud information from the coordinate. FA consists of a series of learnable sine and cosine functions combined with an MLP, which can be expressed as follows:

$$E_i = \Phi_4(\odot \{(\gamma_2 \cdot \sin(2^l \pi S_i \cdot \gamma_1)), (\gamma_2 \cdot \cos(2^l \pi S_i \cdot \gamma_1))\}_{l=0}^L) \quad (3)$$

where E_i is the high-frequency feature of S_i after enhancement learning, γ_1, γ_2 are learnable parameters. L represents the number of frequency mapping layers, which is set to 10 based on experiments, and Φ_4 is the MLP. For each point x_i in S_i , we use the L layers adaptive sine and cosine mapping functions to lift its 3D coordinates $x_i = [x, y, z] \in \mathbb{R}^{1 \times 3}$ into a high-dimensional feature vector $e_{x_i} \in \mathbb{R}^{1 \times (2 \times 3 \times L)}$. In this process, γ_1, γ_2 and Φ_4 dynamically adjust the mapping scale in a data-driven manner, balancing the contributions of different frequency components to the gradients. FA amplifies coordinate differences with adaptive high-dimensional Fourier mapping, improving the network's ability to capture high-frequency details and providing more accurate results in complex geometries.

Afterward, as illustrated in the Fig. 2(c), by fusing the outputs of the coordinate feature f_i and frequency branches E_i , we derive the fused frequency-aware feature F_i of hierarchical layer S_i .

$$F_i = \Phi_5(\odot(f_{i1}, f_{i2}, f_{i3}, E_i)) \quad (4)$$

where f_{i1}, f_{i2}, f_{i3} is processed patch graph ifeatures of sub-patch S_i , Φ_5 is MLP layer.

3.4. Frequency-Aware Hierarchical Geometry

Numerous prior studies have demonstrated that multi-scale architectures yield significant performance gains for normal estimation networks. However, the common strategy is to integrate different scale features along with downsampling and channel transformations, which may inadvertently lead to the loss of fine-grained details during forward propagation. As shown in the Fig. 2(a) and (c), we design a parallel hierarchical architecture that guarantees the intrinsic multi-scale geometric features at each level are maintained as the network progresses. To fuse features across different levels, we introduce an AHFF that dynamically integrates geometric features across different hierarchical scales.

$$M_S = \Phi_6(\Phi_7 \left(\bigoplus_{s=0}^S (Attn \cdot F_s) \right) + \Phi_8(\odot)(F_1, F_2, F_3)) \quad (5)$$

$$Attn = \Phi_9(\text{MAX}\{\bigoplus_{i=0}^S F_i\}) \quad (6)$$

where M_S denote the final aggregated multi-scale features, S represent the total hierarchical level index, and F_i denote the local features at scale S_i , $\Phi_6, \Phi_7, \Phi_8, \Phi_9$ is MLP layers. Within the Φ_6 , we perform a downsampling operation on the obtained hierarchical features to enable precise local feature extraction in the subsequent normal estimation stage.

3.5. Weighted Normal Prediction

As shown in the Fig. 2(d), our network predicts a patch points weight W_n to ensure that only the points genuinely beneficial for the final normal prediction receive larger parameter coefficients, ensure the network can learn to differentiate the correct local surface variations. Finally, we leverage attention mechanisms to predict normals from the final hierarchical geometry feature M_S .

$$n_q = \Phi_{13}(\text{MAX}\{\Phi_{12}(M_{SW})\} \otimes (\Phi_{11}(M_{SW}))) \quad (7)$$

where $M_{SW} = \Phi_{10}(W_n \cdot M_S)$, $\text{MAX}\{\cdot\}$ represents maxpooling features over downsampled patch.

3.6. Loss Function

Our loss function consists of several components. The first is a regularization loss function for Z-direction correction, which constrains the training of the QSTN rotation matrix R .

$$\mathcal{L}_1 = \|n_{gt} R \times z\|, \quad (8)$$

where $z = (0, 0, 1)$.

To constrain normal prediction both geometrically and mathematically, we simultaneously compute the sine distance and the squared Euclidean distance between the predicted normals n_q^{pre} and the ground truth normals n_q^{gt}

$$\mathcal{L}_2 = (\|n_q^{pre} \times n_q^{gt}\| + \min(\|n_q^{pre} - n_q^{gt}\|^2, \|n_q^{pre} + n_q^{gt}\|^2)) \quad (9)$$

Additionally, we evaluate the continuity of the normals between

Table 1: Normal estimation RMSE ↓ comparison on PCPNet, FamousShape and SceneNN datasets. The best results are marked in **Red, and the second-best are marked in **Blue**.**

Methods	year	PCPNet Dataset							FamousShape Dataset							SceneNN Dataset		
		Noise				Density		AVG	Noise				Density		AVG	Clean	Noise	AVG
		None	0.12%	0.60%	1.20%	Strip.	Grad.		None	0.12%	0.60%	1.20%	Strip.	Grad.				
PCA	1992	12.29	12.87	18.38	27.52	13.66	12.81	16.26	19.90	20.60	31.33	45.00	19.84	18.54	25.87	15.93	16.32	16.12
Jet	2005	12.35	12.84	18.33	27.68	13.39	13.13	16.29	20.11	20.57	31.34	45.19	1.82	18.69	25.79	15.17	15.59	15.38
PCPNet	2018	9.64	11.51	18.27	22.84	11.73	13.46	14.58	18.47	21.07	32.60	39.93	18.14	19.50	24.95	20.86	21.40	21.13
Nesti-Net	2019	7.06	10.24	17.77	22.31	8.64	8.95	12.50	11.60	16.80	31.61	39.22	12.33	11.77	20.55	13.01	15.19	14.10
DeepFit	2020	6.51	9.21	16.73	23.12	7.92	7.31	11.80	11.21	16.39	29.84	39.95	11.84	10.54	19.96	10.33	13.07	11.70
AdaFit	2021	5.19	9.05	16.45	21.94	6.01	5.90	10.76	9.09	15.78	29.78	38.74	8.52	8.57	18.41	8.39	12.85	10.62
GraphFit	2022	5.21	8.96	16.12	21.71	6.30	5.86	10.69	8.91	15.73	29.37	38.67	9.10	8.62	18.40	8.39	12.85	10.62
Hsurf-Net	2022	4.17	8.78	16.25	21.61	4.98	4.86	10.11	7.59	15.64	29.43	38.54	7.63	7.40	17.70	7.55	12.33	9.89
NeAF	2023	4.20	9.25	16.35	21.74	4.89	4.88	10.22	7.67	15.67	29.75	38.76	7.22	7.47	17.76	-	-	-
SHSnet	2023	3.95	8.55	16.13	21.53	4.91	4.67	9.96	7.41	15.34	29.33	38.56	7.74	7.28	17.61	7.93	12.40	10.17
MSECNet	2023	3.84	8.74	16.10	21.05	4.34	4.51	9.76	6.73	15.52	29.19	38.06	6.68	6.70	17.15	6.94	11.66	9.30
CMGNet	2024	3.86	8.45	16.08	21.89	4.85	4.45	9.93	7.07	14.83	29.04	38.93	7.43	7.02	17.39	7.64	11.82	9.73
Ours	-	3.25	8.37	16.12	21.10	3.82	3.89	9.43	6.55	14.88	29.37	38.82	6.46	6.38	17.08	7.48	11.65	9.56

the query point q 's neighbor points x_i , with weight w_n .

$$\mathcal{L}_3 = \frac{1}{N} \sum_{i=1}^N w_n (||n_{x_i}^{pre} \times n_{x_i}^{gt}|| + \min(||n_{x_i}^{pre} - n_{x_i}^{gt}||^2, ||n_{x_i}^{pre} + n_{x_i}^{gt}||^2)) \quad (10)$$

where $n_{x_i}^{pre}$ and $n_{x_i}^{gt}$ denote the predicted normal of the neighbor points and the ground truth normal, respectively.

To ensure that the network can effectively identify the points that genuinely contribute to the final normal estimation stage, we use a weight loss to supervise w_n

$$\mathcal{L}_4 = \frac{1}{M} \sum_{i=1}^M (w_{ni}^{pre} - w_{ni}^{gt})^2 \quad (11)$$

where M represents the number of nearest neighbors for query point q after downsampling, w_{ni}^{pre} and w_{ni}^{gt} denote the predicted weight and the ground truth weight of neighbor point x_i , $w_{ni}^{gt} = \exp(-(x_i \cdot n_q^{gt})^2 / \delta^2)$ and $\delta = \max(0.05^2, 0.3 \sum_{i=1}^M (x_i \cdot n_q^{gt})^2 / M)$ [ZCZ*22a].

Finally, the complete loss function is defined as follows

$$\mathcal{L} = \lambda_1 \mathcal{L}_1 + \lambda_2 \mathcal{L}_2 + \lambda_3 \mathcal{L}_3 + \lambda_4 \mathcal{L}_4 \quad (12)$$

where $\lambda_1 = 0.5$, $\lambda_2 = 0.1$, $\lambda_3 = 0.2$, $\lambda_4 = 1$ are the weighting factors determined based on experiment.

4. Experiment

4.1. Datasets

Consistent with recent sota work, we first evaluate our method on the synthetic datasets PCPNet [GKOM18] and FamousShape [LFS*23], each cloud contains 100k points. The point cloud is augmented by adding low, medium, and high noise with standard deviations of 0.12%, 0.6%, and 1.2%. Additionally, density variations (stripes and gradients) are also included. We further evaluated our method on real-world indoor dataset SceneNN [HPN*16] that contains clean and noisy (0.3%) clouds with 100k points in each scene to demonstrate its generalizability.

4.2. Implementation

We exclusively employ the PCPNet dataset for training, following the same experimental setup (including the train-test split and augmentation) described in [LFS*23]. The local patch size N is set to 700 for each randomly selected query point. The sub-patch sizes are set to {700, 350, 150}, respectively, with corresponding KNN scales of {16, 16, 10}. For training, we use the Adam optimizer with an initial learning rate of 5×10^{-4} , which decays following a cosine schedule over 1000 epochs with a batch size of 128. All experiments were conducted on a single RTX 4090 GPU. We use the angle Root Mean Squared Error (RMSE) to evaluate the performance.

4.3. Performance Comparison

To validate the effectiveness of FAHNet we first conducted comprehensive comparative experiments with several classical methods and latest learning-based approaches, including the following: Jet [CP05], PCA [HDD*92], PCPNet [GKOM18], Nesti-Net [BSLF19], DeepFit [BSG20], AdaFit [ZLD*21], GraphFit [LZW*22], Hsurf-Net [LLC*22], NeAF [LZM*23], SHS-Net [LFS*23], MSECNet [XLW*23], CMGNet [WZL*24]. **Table 1** presents a comprehensive statistical summary of all compared approaches evaluated on the PCPNet and FamousShape datasets. Notably, our proposed method achieves the highest accuracy on both synthetic datasets, particularly under low-noise conditions, demonstrating a significant performance improvement. In **Fig. 4** we present visual comparison results for the synthetic datasets PCPNet and FamousShape dataset under varying input conditions. Our method consistently yields markedly lower prediction errors in sharp regions and complex structures compared to alternative approaches, underscoring its superior effectiveness in capturing high-frequency variations in point clouds.

We also conducted experiments on the real-world scanning dataset SceneNN [HPN*16] to evaluate the generalization capability of our method. In **Table 1**, our FAHNet achieved second-optimal

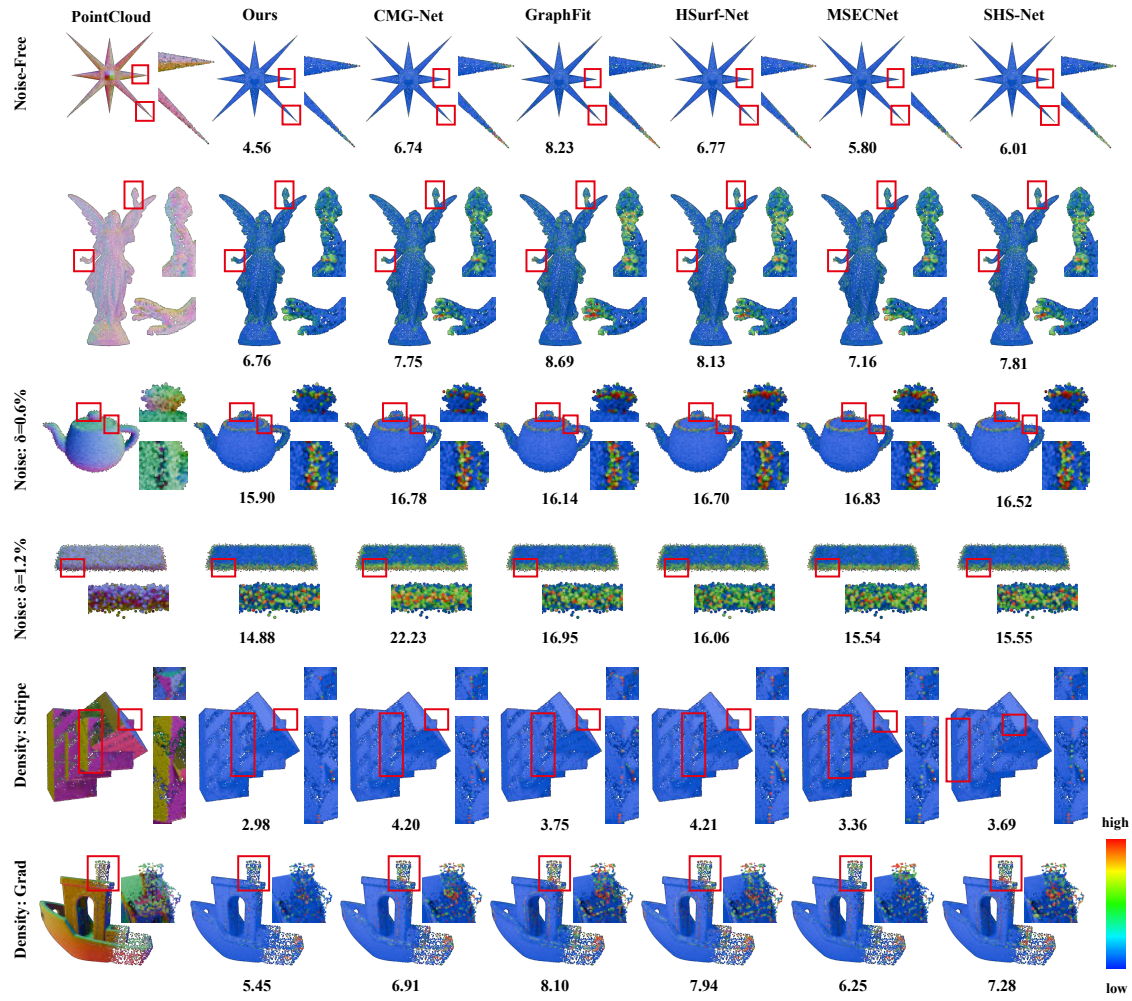


Figure 4: Visual comparisons of normal estimation errors on the synthetic datasets PCPNet and FamousShape with clean, noise, and density variations input. The clouds are rendered with per-point angle error.

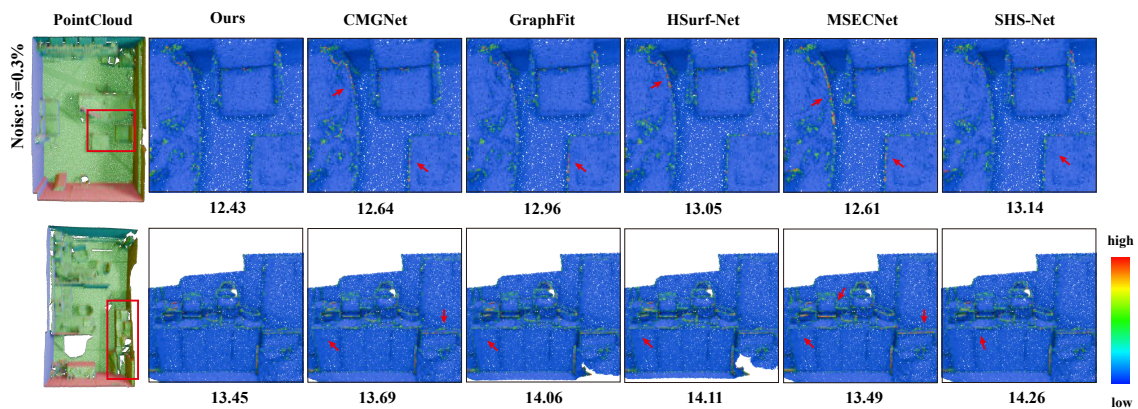


Figure 5: Visual comparisons of normal estimation errors on the real-world dataset SceneNN. The clouds are rendered with per-point angle error.

Table 2: Normal estimation ablation studies on PCPNet dataset with the (a) Frequency-Aware Hierarchical Geometry (FAHG); (b) Attentional Hierarchical Feature Fusion (AHFF); (c) Frequency-Aware Geometry Enhancement (FA); (d) Loss function. The best results are marked in Red.

Ablation Studies		PCPNet Dataset						
		Noise				Density		AVG
		None	0.12%	0.60%	1.20%	Striped	Gradient	
(a)	w/o Attentional Hierarchical Feature Fusion	3.87	8.49	16.26	21.53	4.77	4.59	9.92
(b)	w/o Frequency-Aware Geometry Enhancement	3.42	8.71	16.59	21.71	4.12	4.22	9.79
(c)	w/o Frequency-Aware Hierarchical Geometry (S=1)	3.47	8.34	16.16	21.07	4.16	4.10	9.55
	S=2	3.36	8.37	16.11	21.12	3.98	4.07	9.50
	S=4	3.44	8.57	16.26	21.32	4.15	4.19	9.65
	w/o Weight	3.49	8.33	16.14	21.13	4.00	4.02	9.52
(d)	w/o Sin loss	3.58	8.49	16.15	21.23	4.29	4.14	9.65
	w/o Squared Euclidean Distance loss	3.23	8.37	16.42	21.53	3.94	3.97	9.58
	w/o Neighbor loss	3.30	8.48	16.24	21.29	3.95	3.99	9.54
	w/o QSTN loss	3.58	8.38	16.18	21.08	4.23	4.06	9.58
Full method		3.25	8.37	16.12	21.10	3.82	3.89	9.43

performance, trailing the best result by only 2.7%. However, it is noteworthy that our method uses just 2.3M parameters—less than one quarter of MSECNet’s 10.4M. Consistent with the synthetic datasets, we also present visual comparison results on the real-world dataset SceneNN in Fig. 5. The results demonstrate that our method better understands the edges of sharp objects in real-world scenes, even with noise condition.

To further demonstrate the capability of our method in capturing complex geometric features, we conducted experiments on models with highly intricate geometries. The experiment compares the latest techniques, including non-learning methods iPSR [HWW*22], WNNC [LSL24] and learning-based methods CMG [WZL*24], MSECNet [XLW*23], SHS-Net [LFS*23]. For both of non-learning methods, we use the recommended parameters setting in their official implementations. Each point cloud comprises 100k points, randomly sampled from triangular meshes in the Thingi10K dataset [ZJ16]. The ground-truth normals for each point are the normals of the corresponding triangles on which the points reside. As shown in Fig. 7. All methods tend to make most errors on sharp edges, thin-sheet structures, and acute chamfers. However, by more effectively capturing high-frequency variations, our approach provides the most accurate normal estimates in these particularly difficult areas with large curvature changes.

4.4. Ablation Studies

To systematically evaluate the contribution of each architectural component in FAHNet, we conduct comprehensive ablation studies as follows. The results are presented in Table 2.

(a) **Attentional Hierarchical Feature Fusion.** The proposed AHFF mechanism improves accuracy under all input conditions compared to naive downsampling and concatenation fusion operations. This demonstrates that AHFF can effectively distill useful features across different hierarchical layers.

(b) **Frequency-Aware Hierarchical Geometry.** We analyze the influence of FAHG by progressively increasing the number of

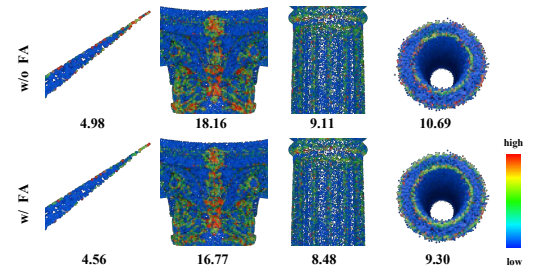


Figure 6: Visualization of the FA module’s effects. Including spikes, complex topologies, chamfers, sharp edges, and noisy input conditions.

FALFE layers from $S = 1$ to $S = 4$. Besides, other entries in Table 2 were conducted with $S = 3$. Experimental results reveal that the three-layer architecture achieves optimal performance, demonstrating that our framework enables effective multi-scale geometric pattern learning while avoiding over-decomposition artifacts.

Table 3: Ablation studies on FA with different layer setups, average RMSE on PCPNet is reported.

	w/o FA	w/ FA
S=1	9.98	9.55
S=2	9.80	9.50
S=4	9.84	9.65
S=3(Full method)	9.79	9.43

(c) **Frequency-Aware Geometry Enhancement.** Compared to architectures without the FA module, it significantly elevates accuracy across various input conditions, especially under noisy input. This demonstrates that FA enables networks to learn high-frequency variation in point clouds robustly. Moreover, the visual comparison given in Fig. 6 confirm that. Additionally, to explore the interaction between FA and the coordinate branch, we also tested FA under dif-

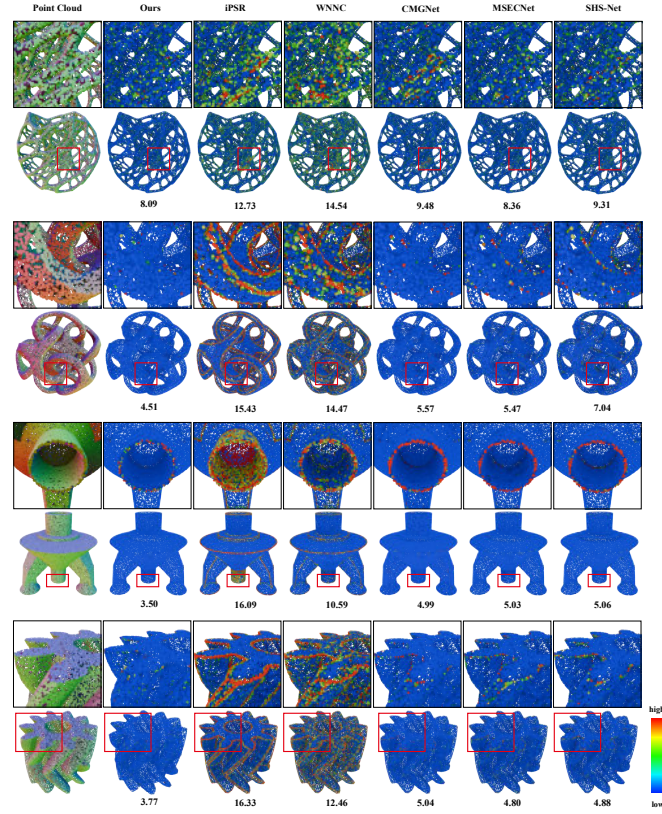


Figure 7: Visual comparison of unoriented normal errors on complex model with highly intricate geometries. The clouds are rendered with per-point angle error.

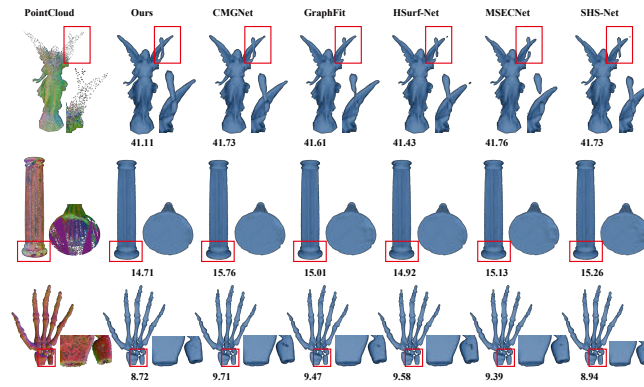


Figure 8: Surface Reconstruction Comparison.

ferent layer setups in **Table 3**. Regardless of the layer configuration, incorporating the FA consistently yields significant improvements in model performance, demonstrating the effectiveness of FA.

(d) Loss function. In addition, we evaluated various combinations of loss functions and identified the optimal configuration. After incorporating a Euclidean regression loss component with physics-informed sine normal loss, the model's robustness to noisy inputs has significantly improved.

4.5. Application

The proposed normal estimation framework demonstrates inherent compatibility with downstream surface reconstruction pipelines. Leveraging our predicted oriented normals as geometric priors, we implement Poisson surface reconstruction [KBH06] to generate watertight meshes from raw point clouds. As demonstrated in **Fig. 8** with Hausdorff Distance ($\times 10^{-3}$), our method achieves superior reconstruction fidelity compared to competing approaches. Even with

incomplete and sparse point-cloud inputs, the surface reconstructed with our normal exhibits the fewest artifacts.

5. Conclusion

In this work, we propose FAHNet, an end-to-end approach for accurate and robust point cloud normal estimation, addressing poor prediction in areas with strong features. By introducing a novel hierarchical architecture incorporating a frequency-aware module, our method enhances prediction accuracy on both synthetic and real-world datasets. However, our approach also exhibits some limitations, performance under noisy conditions still leaves room for improvement, pointing to a promising direction for future research.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 62372152 and the Fundamental Research Funds for the Central Universities of China (Grant No.PA2025GDSK0035).

References

- [AB98] AMENTA N., BERN M.: Surface reconstruction by voronoi filtering. In *Proceedings of the Fourteenth Annual Symposium on Computational Geometry* (New York, NY, USA, 1998), SCG '98, Association for Computing Machinery, p. 39–48. URL: <https://doi.org/10.1145/276884.276889>, doi:10.1145/276884.276889. 3
- [ACSTD07] ALLIEZ P., COHEN-STEINER D., TONG Y., DESBRUN M.: Voronoi-based variational reconstruction of unoriented point sets. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing* (Goslar, DEU, 2007), SGP '07, Eurographics Association, p. 39–48. 3
- [ASL*17] AROUDI S., SEEMANN P., LANGGUTH F., GUTHE S., GOESELE M.: Visibility-consistent thin surface reconstruction using multi-scale kernels. *ACM Trans. Graph.* 36, 6 (Nov. 2017). URL: <https://doi.org/10.1145/3130800.3130851>, doi:10.1145/3130800.3130851. 3
- [BM12] BOULCH A., MARLET R.: Fast and robust normal estimation for point clouds with sharp features. *Comput. Graph. Forum* 31, 5 (Aug. 2012), 1765–1774. URL: <https://doi.org/10.1111/j.1467-8659.2012.03181.x>, doi:10.1111/j.1467-8659.2012.03181.x. 3
- [BM16] BOULCH A., MARLET R.: Deep learning for robust normal estimation in unstructured point clouds. In *Proceedings of the Symposium on Geometry Processing* (Goslar, DEU, 2016), SGP '16, Eurographics Association, p. 281–290. 3
- [BSG20] BEN-SHABAT Y., GOULD S.: Deepfit: 3d surface fitting via neural network weighted least squares. In *Computer Vision – ECCV 2020* (Cham, 2020), Vedaldi A., Bischof H., Brox T., Frahm J.-M., (Eds.), Springer International Publishing, pp. 20–34. 1, 3, 6
- [BSLF19] BEN-SHABAT Y., LINDENBAUM M., FISCHER A.: Nesti-net: Normal estimation for unstructured 3d point clouds using convolutional neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2019). 3, 6
- [CO18] CHE E., OLSEN M. J.: Multi-scan segmentation of terrestrial laser scanning data based on normal variation analysis. *ISPRS Journal of Photogrammetry and Remote Sensing* 143 (2018), 233–248. *ISPRS Journal of Photogrammetry and Remote Sensing Theme Issue “Point Cloud Processing”*. URL: <https://www.sciencedirect.com/science/article/pii/S0924271618300248>, doi:<https://doi.org/10.1016/j.isprsjprs.2018.01.019>. 1
- [CP05] CAZALS F., POUGET M.: Estimating differential quantities using polynomial fitting of osculating jets. *Comput. Aided Geom. Des.* 22, 2 (Feb. 2005), 121–146. 3, 6
- [DYW*23] DU H., YAN X., WANG J., XIE D., PU S.: Rethinking the approximation error in 3d surface fitting for point cloud normal estimation. pp. 9486–9495. doi:10.1109/CVPR52729.2023.00915. 3
- [GG07] GUENNEBAUD G., GROSS M.: Algebraic point set surfaces. *ACM Trans. Graph.* 26, 3 (July 2007), 23–es. URL: <https://doi.org/10.1145/1276377.1276406>, doi:10.1145/1276377.1276406. 3
- [GKOM18] GUERRERO P., KLEIMAN Y., OVSJANIKOV M., MITRA N. J.: PCPNet: Learning local shape properties from raw point clouds. *Computer Graphics Forum* 37, 2 (2018), 75–85. doi:10.1111/cgf.13343. 1, 2, 3, 4, 6
- [GRAS*21] GARRIDO D., RODRIGUES R., AUGUSTO SOUSA A., JACOB J., CASTRO SILVA D.: Point cloud interaction and manipulation in virtual reality. In *2021 5th International Conference on Artificial Intelligence and Virtual Reality (AIVR)* (New York, NY, USA, 2021), AIVR 2021, Association for Computing Machinery, p. 15–20. URL: <https://doi.org/10.1145/3480433.3480437>, doi:10.1145/3480433.3480437. 1
- [GZXS24] GU S., ZHANG M., XIAO Q., SHI W.: Cascaded matching based on detection box area for multi-object tracking. *Knowledge-Based Systems* 299 (2024), 112075. URL: <https://www.sciencedirect.com/science/article/pii/S0950705124007093>, doi:<https://doi.org/10.1016/j.knosys.2024.112075>. 1
- [HDD*92] HOPPE H., DE ROSE T., DUCHAMP T., McDONALD J., STUETZLE W.: Surface reconstruction from unorganized points. *SIGGRAPH Comput. Graph.* 26, 2 (July 1992), 71–78. URL: <https://doi.org/10.1145/142920.134011>, doi:10.1145/142920.134011. 1, 3, 6
- [HFZ*24] HUANG G., FANG Q., ZHANG Z., LIU L., FU X.-M.: Stochastic normal orientation for point clouds. 3
- [HLZ*09] HUANG H., LI D., ZHANG H., ASCHER U., COHEN-OR D.: Consolidation of unorganized point clouds for surface reconstruction. *ACM Trans. Graph.* 28, 5 (Dec. 2009), 1–7. URL: <https://doi.org/10.1145/1618452.1618522>, doi:10.1145/1618452.1618522. 1, 3
- [HPN*16] HUA B.-S., PHAM Q.-H., NGUYEN D. T., TRAN M.-K., YU L.-F., YEUNG S.-K.: Scenenn: A scene meshes dataset with annotations. In *International Conference on 3D Vision (3DV)* (2016). 2, 6
- [HS19] HASHIMOTO T., SAITO M.: Normal estimation for accurate 3d mesh reconstruction with point cloud model incorporating spatial structure. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops* (June 2019). 3
- [HWW*22] HOU F., WANG C., WANG W., QIN H., QIAN C., HE Y.: Iterative poisson surface reconstruction (ipsr) for unoriented points. URL: <https://doi.org/10.1145/3528223.3530096>, doi:10.1145/3528223.3530096. 3, 8
- [KBH06] KAZHDAN M., BOLITHO M., HOPPE H.: Poisson surface reconstruction. SGP '06, Eurographics Association, p. 61–70. 1, 9
- [KH13a] KAZHDAN M., HOPPE H.: Screened poisson surface reconstruction. *ACM Trans. Graph.* 32, 3 (July 2013). URL: <https://doi.org/10.1145/2487228.2487237>, doi:10.1145/2487228.2487237. 1
- [KH13b] KAZHDAN M., HOPPE H.: Screened poisson surface reconstruction. *ACM Trans. Graph.* 32, 3 (July 2013). URL: <https://doi.org/10.1145/2487228.2487237>, doi:10.1145/2487228.2487237. 3
- [Lev98] LEVIN D.: The approximation power of moving least-squares. *Math. Comput.* 67, 224 (Oct. 1998), 1517–1531. URL: <https://doi.org/10.1090/S0025-5718-98-00974-0>, doi:10.1090/S0025-5718-98-00974-0. 3

- [LFS*23] LI Q., FENG H., SHI K., GAO Y., FANG Y., LIU Y.-S., HAN Z.: SHS-Net: Learning signed hyper surfaces for oriented normal estimation of point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Los Alamitos, CA, USA, June 2023), IEEE Computer Society, pp. 13591–13600. URL: <https://doi.ieeecomputersociety.org/10.1109/CVPR52729.2023.01306>, 1, 2, 3, 6, 8
- [LLC*22] LI Q., LIU Y.-S., CHENG J.-S., WANG C., FANG Y., HAN Z.: Hsurf-net: normal estimation for 3d point clouds by learning hyper surfaces. In *Proceedings of the 36th International Conference on Neural Information Processing Systems* (Red Hook, NY, USA, 2022), NIPS '22, Curran Associates Inc. 1, 3, 6
- [LLX*24] LIN A., LI J., XIANG Y., BIAN W., PRASAD M.: Normal transformer: Extracting surface geometry from lidar points enhanced by visual semantics. *IEEE Transactions on Intelligent Vehicles PP* (01 2024), 1–11. doi:10.1109/TIV.2024.3363174. 3
- [LLZ*24] LI N., LI X., ZHOU J., JIANG D., LIU J., QIN H.: Geohi-gnn: Geometry-aware hierarchical graph representation learning for normal estimation. *Comput. Aided Geom. Des.* 114 (2024), 102390. URL: <https://api.semanticscholar.org/CorpusID:273278650>. 3
- [LOM19] LENNSEN J. E., OSENDORFER C., MASCI J.: Deep iterative surface normal estimation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), 11244–11253. URL: <https://api.semanticscholar.org/CorpusID:210088975>. 1, 3
- [LSL24] LIN S., SHI Z., LIU Y.: Fast and globally consistent normal orientation based on the winding number normal consistency. URL: <https://doi.org/10.1145/3687895>, doi:10.1145/3687895. 3, 8
- [LZM*23] LI S., ZHOU J., MA B., LIU Y.-S., HAN Z.: Neaf: Learning neural angle fields for point normal estimation. In *Proceedings of the AAAI conference on artificial intelligence* (2023), vol. 37, pp. 1396–1404. 3, 6
- [LZW*22] LI K., ZHAO M., WU H., YAN D.-M., SHEN Z., WANG F.-Y., XIONG G.: Graphfit: Learning multi-scale graph-convolutional representation for point cloud normal estimation. In *Computer Vision – ECCV 2022* (Cham, 2022), Avidan S., Brostow G., Cissé M., Farinella G. M., Hassner T., (Eds.), Springer Nature Switzerland, pp. 651–667. 1, 3, 6
- [MN03] MITRA N. J., NGUYEN A.: Estimating surface normals in noisy point cloud data. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry* (New York, NY, USA, 2003), SCG '03, Association for Computing Machinery, p. 322–328. URL: <https://doi.org/10.1145/777792.777840>, doi:10.1145/777792.777840. 1, 3
- [MOG11] MERIGOT Q., OVSJANIKOV M., GUIBAS L. J.: Voronoi-based curvature and feature estimation from point clouds. *IEEE Transactions on Visualization and Computer Graphics* 17, 6 (June 2011), 743–756. URL: <https://doi.org/10.1109/TVCG.2010.261>, doi:10.1109/TVCG.2010.261. 3
- [MOW24] MARIN D., OHRHALLINGER S., WIMMER M.: Parameter-free connectivity for point clouds. In *VISIGRAPP* (2024). URL: <https://api.semanticscholar.org/CorpusID:268234891>. 3
- [MST*20] MILDENHALL B., SRINIVASAN P. P., TANCİK M., BARRON J. T., RAMAMOORTHY R., NG R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV* (2020). 5
- [QSMG16] QI C. R., SU H., MO K., GUIBAS L. J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593* (2016). 3
- [RBA*19] RAHAMAN N., BARATIN A., ARPIT D., DRAXLER F., LIN M., HAMPRECHT F., BENGIO Y., COURVILLE A.: On the spectral bias of neural networks. In *Proceedings of the 36th International Conference on Machine Learning* (09–15 Jun 2019), Chaudhuri K., Salakhutdinov R., (Eds.), vol. 97 of *Proceedings of Machine Learning Research*, PMLR, pp. 5301–5310. URL: <https://proceedings.mlr.press/v97/rahaman19a.html>. 2, 5
- [TSM*20] TANCİK M., SRINIVASAN P. P., MILDENHALL B., FRIDOVICH-KEIL S., RAGHAVAN N., SINGHAL U., RAMAMOORTHY R., BARRON J. T., NG R.: Fourier features let networks learn high frequency functions in low dimensional domains. *NIPS '20*, Curran Associates Inc. 5
- [WQF19] WU W., QI Z., FUXIN L.: Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 9621–9630. 1
- [WWR22] WIMBAUER F., WU S., RUPPRECHT C.: De-rendering 3d objects in the wild. In *CVPR* (2022). 1
- [WZL*24] WU Y., ZHAO M., LI K., QUAN W., YU T., YANG J., JIA X., YAN D.-M.: Cmg-net: robust normal estimation for point clouds via chamfer normal distance and multi-scale geometry. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence* (2024), AAAI'24/IAAI'24/EAAI'24, AAAI Press. URL: <https://doi.org/10.1609/aaai.v38i6.28434>, doi:10.1609/aaai.v38i6.28434. 1, 2, 3, 6, 8
- [XLW*23] XIU H., LIU X., WANG W., KIM K.-S., MATSUOKA M.: Msecnet: Accurate and robust normal estimation for 3d point clouds by multi-scale edge conditioning. *arXiv:2308.02237* (2023). 1, 2, 3, 6, 8
- [YCSL24] YANG Z., CHEN L., SUN Y., LI H.: Visual point cloud forecasting enables scalable autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2024). 1
- [ZCL*19] ZHANG J., CAO J., LIU X., CHEN H., LI B., LIU L.: Multi-normal estimation via pair consistency voting. *IEEE Transactions on Visualization and Computer Graphics* 25, 4 (Apr. 2019), 1693–1706. URL: <https://doi.org/10.1109/TVCG.2018.2827998>, doi:10.1109/TVCG.2018.2827998. 3
- [ZCZ*22a] ZHANG J., CAO J.-J., ZHU H.-R., YAN D.-M., LIU X.-P.: Geometry guided deep surface normal estimation. *Comput. Aided Des.* 142, C (Jan. 2022). URL: <https://doi.org/10.1016/j.cad.2021.103119>, doi:10.1016/j.cad.2021.103119. 1, 3, 6
- [ZCZ*22b] ZHOU H., CHEN H., ZHANG Y., WEI M., XIE H., WANG J., LU T., QIN J., ZHANG X.-P.: Refine-net: Normal refinement neural network for noisy point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence PP* (01 2022), 1–1. doi:10.1109/TPAMI.2022.3145877. 1, 3
- [ZHLL19] ZHOU J., HUANG H., LIU B., LIU X.: Normal estimation for 3d point clouds via local plane constraint and multi-scale selection, 10 2019. doi:10.48550/arXiv.1910.08537. 3
- [ZJ16] ZHOU Q., JACOBSON A.: Thingi10k: A dataset of 10,000 3d-printing models. *arXiv preprint arXiv:1605.04797* (2016). 8
- [ZJW*21] ZHOU J., JIN W., WANG M., LIU X., LI Z., LIU Z.: Improvement of normal estimation for pointclouds via simplifying surface fitting. *ArXiv abs/2104.10369* (2021). URL: <https://api.semanticscholar.org/CorpusID:233324529>. 3
- [ZLD*21] ZHU R., LIU Y., DONG Z., WANG Y., JIANG T., WANG W., YANG B.: AdaFit: Rethinking learning-based normal estimation on point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (October 2021), pp. 6118–6127. 1, 3, 6