

Article

Research on a 3D Point Cloud Map Learning Algorithm Based on Point Normal Constraints

Zhao Fang ¹, Youyu Liu ^{1,*} , Lijin Xu ², Mahamudul Hasan Shahed ¹ and Liping Shi ²

¹ School of Mechanical and Automotive Engineering, Anhui Polytechnic University, Wuhu 241000, China; 2220110177@stu.ahpu.edu.cn (Z.F.); shahedhasan201115@gmail.com (M.H.S.)

² Anhui Gongchuang Industrial Robot Innovation Center Co., Ltd., Wuhu 241100, China; xulijin@anhui-at.com (L.X.); shiliping@wh-robot.cn (L.S.)

* Correspondence: liuyyu@ahpu.edu.cn

Abstract: Laser point clouds are commonly affected by Gaussian and Laplace noise, resulting in decreased accuracy in subsequent surface reconstruction and visualization processes. However, existing point cloud denoising algorithms often overlook the local consistency and density of the point cloud normal vector. A feature map learning algorithm which integrates point normal constraints, Dirichlet energy, and coupled orthogonality bias terms is proposed. Specifically, the Dirichlet energy is employed to penalize the difference between neighboring normal vectors and combined with a coupled orthogonality bias term to enhance the orthogonality between the normal vectors and the subsurface, thereby enhancing the accuracy and robustness of the learned denoising of the feature maps. Additionally, to mitigate the effect of mixing noise, a point cloud density function is introduced to rapidly capture local feature correlations. In experimental findings on the anchor public dataset, the proposed method reduces the average mean square error (MSE) by 0.005 and 0.054 compared to the MRPCA and NLD algorithms, respectively. Moreover, it improves the average signal-to-noise ratio (SNR) by 0.13 DB and 2.14 DB compared to MRPCA and AWLOP, respectively. The proposed algorithm enhances computational efficiency by 27% compared to the RSLDM method. It not only removes mixed noise but also preserves the local geometric features of the point cloud, further improving computational efficiency.

Keywords: point cloud denoising; point normal constraints; Dirichlet energy; Gaussian noise; Laplace noise



Citation: Fang, Z.; Liu, Y.; Xu, L.; Shahed, M.H.; Shi, L. Research on a 3D Point Cloud Map Learning Algorithm Based on Point Normal Constraints. *Sensors* **2024**, *24*, 6185. <https://doi.org/10.3390/s24196185>

Academic Editors: Roberto Teti, Peng Cao and Junxing Zheng

Received: 22 May 2024

Revised: 15 September 2024

Accepted: 20 September 2024

Published: 24 September 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Point clouds find extensive applications across various domains, including 3D modeling and autonomous driving [1–3]. Typically, point cloud data are acquired through two methods [4,5]: 3D scanners and stereo matching algorithms. These methods can introduce significant noise due to matching inaccuracies. The alternative approach employs devices like laser scanners [6], structured light scanners [7], or Lidar [8] for quick real-time point cloud data acquisition. However, point clouds often suffer from noise stemming from both hardware and software sources [9]. This noise not only distorts the underlying structure of the point cloud, impeding surface reconstruction [10] and subsequent tasks, but also increases the presence of irrelevant data and reduces the accuracy of feature extraction, such as point cloud curvature and normal vectors [11].

Noise reduction in point clouds has emerged as a crucial area of focus in processing three-dimensional geometric data. Zeng et al. [12] introduced a method employing graph Laplacian regularization (GLR) for noise reduction. This method extends the low-dimensional manifold model utilized for two-dimensional images to the point cloud surface, treating it as a discrete observation on the manifold. It employs a local region-based graph Laplacian regularizer to approximate the manifold dimension defined in the continuous domain and proposes a novel discrete distance measure to assess the similarity between

two equally sized local regions. This measure is then used to construct a noise-resistant graph and facilitate the noise reduction process. Hu et al. [13] proposed a feature map learning paradigm to deduce the underlying graph structure of the point cloud for noise reduction. Unlike the manual adjustment of edge weight parameters, their approach optimizes these parameters strictly through feature measurement learning when the available signal is smooth relative to the graph, thus achieving accurate and rapid noise reduction of the point cloud.

The above algorithms have achieved good results for the additional noise generated on the surface of objects, but they are not suitable for actual mixed noise far away from object noise points. Mixed noise mainly includes Gaussian noise and Laplacian noise. Gaussian noise is usually caused by sensor thermal noise, while Laplacian noise can originate from hardware failures or environmental interference. These types of noise are extremely common in practical applications, and each has different statistical and distribution characteristics, so removing mixed noise points remains challenging.

Therefore, this study proposes a denoising algorithm for Gaussian noise and Laplacian noise to address this challenge. Its model is shown in Figure 1.

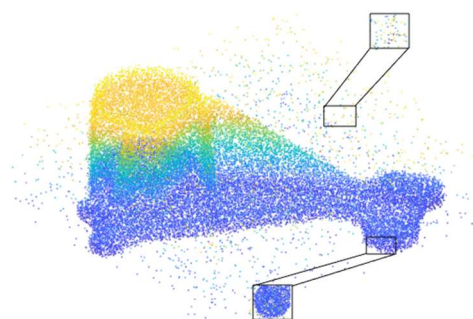


Figure 1. A point cloud model with Gaussian noise and Laplacian noise.

This paper presents a novel feature map learning denoising algorithm with point normal constraints to address the aforementioned issues. The key innovations and contributions include:

- (1) The integration of Dirichlet energy with the coupling orthogonality deviation term to improve the accuracy and robustness of feature map learning denoising;
- (2) The introduction of a density function to better illustrate the correlation between the local features of the point cloud, preserving the fundamental structure of the denoised point cloud;
- (3) The design of denoising algorithms for Gaussian noise and Laplace noise, thereby eliminating both types of noise simultaneously.

2. Related Work

Currently, noise reduction techniques for point cloud data can be classified into six main categories: sparsity-based methods, non-local methods, techniques employing moving least squares (MLS), approaches based on local optimal projection (LOP), graph-based methods, and those utilizing deep learning. The sparsity method assumes that point cloud data have a sparse representation in a certain transformation domain, thus achieving denoising. It can effectively remove noise and preserve details, but its computational complexity is high and the processing efficiency of large point cloud data is low. Non-local methods perform well in preserving details and edges by identifying and utilizing similar patterns in point clouds for denoising. However, when there is a lot of noise in the point cloud, the accuracy of identifying similar patterns decreases, which affects the denoising effect. The MLS method utilizes moving least squares fitting technology to smooth point cloud data, which is suitable for removing small-scale noise and can smooth the surface while preserving most geometric features. However, its ability to handle complex geometric structures and large-scale noise is limited. The LOP method iteratively optimizes and

projects the point cloud onto a more regular surface, effectively removing noise while preserving geometric features. However, the computational cost is high, and in some cases, it may be overly smooth. The graph-based method utilizes graph structure to represent point clouds, and denoising is carried out through graph signal processing techniques, which can effectively handle mixed noise, enhancing denoising accuracy and robustness. However, it requires constructing and processing graph structures, which require high computational resources for very large point cloud data. Deep learning methods learn the features of point cloud denoising by training neural network models. After training on large-scale datasets, they can achieve a balance between accuracy and efficiency, but require a large amount of annotated data for training and require vast computational resources.

2.1. Methods Based on Sparsity

This method employs sparse representation aligned with the point's normal vector, addressing the global issue via sparse regularization and conducting sparse reconstruction in the same direction. It subsequently adjusts the point's position [14,15] by solving a further global minimization problem, predicated on the local plane assumption. Mattei et al. [16] introduced the mobile robust principal component analysis (MRPCA) method, utilizing a weighted minimization approach to denoise three-dimensional point clouds while preserving sharp features. However, these techniques may yield suboptimal results when local areas with high signal-to-noise ratios produce redundant features, potentially leading to oversmoothing or oversharpening.

2.2. Non-Local Methods

Drawing inspiration from the non-local means (NLM) [17] and BM3D [18] image denoising algorithms, the proposed method leverages the self-similarity of point cloud surface patches for noise reduction. Deschaud et al. [19] utilized the polynomial coefficients of the local moving least squares (MLS) surface as neighborhood descriptors to compute point similarity, thereby introducing a non-local denoising (NLD) algorithm. However, this approach depends on the self-similarity among surface blocks in the point cloud and is marked by high computational complexity.

2.3. Based on the Moving Least Squares (MLS) Method

Alexa et al. [20] established the point cloud surface through a process of moving least squares projection. This involves projecting any noise points to the least squares surface before smoothing the data. Guennebaud et al. [21] introduced algebraic point set surfaces (APSS), a technique that can inadvertently lead to oversmoothing in the denoising effect.

2.4. Methods Based on Local Optimal Projection (LOP)

The local optimum projection (LOP) method aims to minimize noise by projecting a point set onto the natural surface of a point cloud. Huang et al. [22] introduced the weighted local optimum projection (WLOP) algorithm, which enhances the LOP algorithm. While WLOP preserves the point cloud's features, it tends to be computationally inefficient. Additionally, Huang et al. introduced the anisotropic weighted local optimum projection (AWLOP) algorithm [23]. This algorithm is a modification of WLOP, utilizing an anisotropic weighted function to retain the point cloud's sharp features. However, it can generate additional features and smooth the results.

2.5. Graph-Based Methods

The method of image filtering is based on graph filtering for noise reduction. Dinesh et al. [24] addressed various types of additive noise using two fidelity terms via feature graph Laplacian regularization. Schoenenberger et al. [25] constructed a k-nearest neighbor graph on the input point cloud, subsequently creating a convex optimization problem on the graph, regularized by the point cloud gradient. Zhang et al. [26] encoded the similarity between available pixels as edge weights through feature map learning, and established

another corrected similarity between depth images through viewpoint mapping and sparse linear interpolation, achieving posterior denoising of point clouds. Zou et al. [27] extended the low-dimensional manifold model (RSLDM) of two-dimensional images to point cloud surfaces using k-domain partitioning, improved the low dimensional manifold model, and proposed a statistical low-dimensional manifold model. Hu et al. [13] suggested learning feature graphs to optimize edge weights, but they did not consider the constraints of normal vectors, which made it challenging to achieve satisfactory results in complex mixed noise environments.

2.6. Methods Based on Deep Learning

Given the constraints of conventional noise reduction algorithms [28] and the success of deep learning [29] in diverse fields in recent years, Qi et al. [30] introduced a pioneering network, PointNet, to directly process raw point cloud data. This addressed the issue that standard deep neural networks necessitate structured inputs, complicating the training process on unordered point clouds. Consequently, it made noise reduction via deep learning feasible. On this foundation, a plethora of outstanding deep learning-based noise reduction techniques have emerged. Among them, PCN [31] is the first network to specialize in noise reduction using deep learning. However, these methods are restricted by the inherent limitations of computer hardware.

3. Methods

3.1. Normal Point Constraint Feature Map Learning Model

3.1.1. Three-Dimensional Point Cloud and Noise Model

The point cloud is $\mathcal{N} = \{\mathbf{p}_i\}_{i=1}^N$, where the points in the point cloud are $\mathbf{p}_i \in \mathbb{R}^{3 \times N}$. The observation model [27] of the point cloud is shown in Equation (1):

$$\mathbf{N} = \mathbf{U} + \mathbf{G} + \mathbf{L} \quad (1)$$

where $\mathbf{N}$ represents the three-dimensional coordinates of N points affected by observational noise, $\mathbf{U}$ represents the true three-dimensional coordinates of the point cloud, $\mathbf{G}$ represents Gaussian noise, as shown in Equation (2), and $\mathbf{L}$ represents Laplacian noise, as shown in Equation (3):

$$\mathbf{G} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}) \quad (2)$$

where σ represents standard deviation and $\mathbf{I}$ represents an identity matrix. Furthermore, $\mathbf{U}, \mathbf{G}, \mathbf{L} \in \mathbb{R}^{3 \times N}$.

$$\mathbf{L} \sim \mathcal{N}(\phi, b) \quad (3)$$

where ϕ represents the position parameter, and $b^2 = \frac{1}{2}\sigma$ represents the scale parameter. The goal is to restore the observed model $\mathbf{N}$ to the ideal model $\mathbf{U}$ under the interference of Gaussian noise $\mathbf{G}$ and Laplace noise $\mathbf{L}$.

3.1.2. Normal Vector Local Consistency Constraint

To enhance noise removal, the concept of local consistency in point cloud normal vectors is introduced, as depicted in Figure 2. The Dirichlet energy $E_D(\alpha)$ [32] is utilized to penalize the disparities between adjacent normal vectors. This approach ensures the continuity of the point cloud normal vector field, as illustrated in Equation (4):

$$E_D(\alpha) = \frac{1}{2} \sum_i \sum_{j \in N(i)} \|\alpha_i \mathbf{n}_i - \alpha_j \mathbf{n}_j\|^2 r_{ij} \quad (4)$$

where $\mathbf{n}_i = (n_{i1}, n_{i2}, n_{i3})^T$ represents the normal vector on point $\mathbf{p}_i$, $\mathbf{n}_j = (n_{j1}, n_{j2}, n_{j3})^T$ represents the normal vector on point $\mathbf{p}_j$, and $\mathbf{p}_i$ and $\mathbf{p}_j$ represent two adjacent points on the point cloud block. $r_{ij} = \exp(-\frac{\|\mathbf{p}_i - \mathbf{p}_j\|}{\theta^2})$, $\theta = (\frac{Max}{(q_i, q_j) \in \xi}) \frac{\|\mathbf{p}_i - \mathbf{p}_j\|}{2}$.

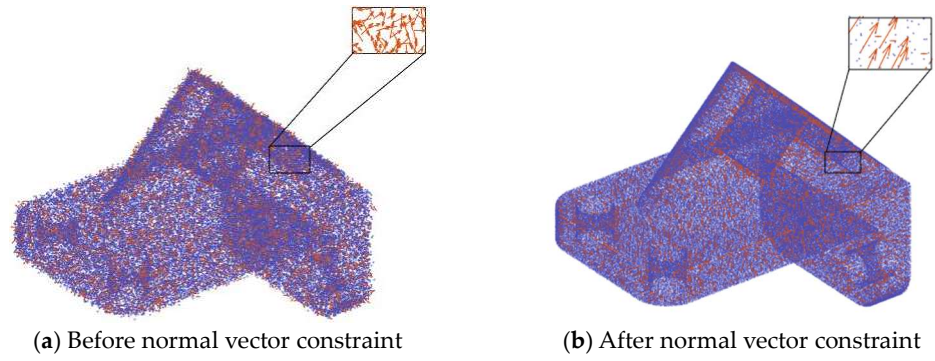


Figure 2. Local consistency constraint of point cloud normal vectors.

The coupling orthogonality deviation item $E_{COD}(\alpha)$ [32] is combined to enhance the orthogonality condition between the normal vector and the bottom surface of point cloud data, as illustrated in Equation (5):

$$E_{COD}(\alpha) = \sum_i \sum_{j \in N(i)} r_{ij} \left[(\alpha_i n_i + \alpha_j n_j)^T \left(\frac{p_i - p_j}{\alpha_{ij}} \right) \right]^2 \quad (5)$$

where $n_i = \alpha_i \bar{n}_i$ and $\alpha_i = (\alpha_1, \alpha_2, \dots, \alpha_m)^T$ are decision vectors.

For point clouds with noisy inputs, the sum of the Dirichlet energy and the coupling orthogonality deviation term $E_{loc}(\alpha)$ is referred to, and the regularizer L_1 is applied to ensure stability, as shown in Equation (6).

$$E_{loc}(\alpha) = E_D(\alpha) + E_{COD}(\alpha) + \|\alpha_i\|_1 \quad (6)$$

From Equations (4) and (5), it can be concluded that $E_D(\alpha)$ and $E_{COD}(\alpha)$ are quadratic functions of $n = (n_1^T, n_2^T, \dots, n_N^T)$. Therefore, the sum of the Dirichlet energy and the coupling orthogonality deviation term can be represented as shown in Equation (7):

$$E_{loc}(\alpha) = n^T P n \quad (7)$$

where P represents a sparse matrix. For denoising a point cloud with noisy input, the constraint of local consistency in the normal vector ensures that its local features are preserved after denoising.

3.1.3. Probability Density Function

Each point in the point cloud, labeled as point c_i , is used to construct a point cloud block by identifying its k nearest neighbors. To meet the self-similarity requirement of the point cloud block, it must be translated and aligned, ensuring that the central point of each block is situated at the origin. The mathematical representation of the point cloud block [13] is provided in Equation (8):

$$V = (TN - C) \quad (8)$$

where $T \in (0, 1)^{(k+1)M \times N}$ is the selection matrix, $C = \{c_i\}_{i=1}^M$, and points are selected from the point cloud N to form M point cloud blocks, each of which contains $k + 1$ points.

To accommodate the complexities of object point cloud denoising, the density function $\zeta = f(c_i)$ is introduced, and Equation (9) is derived as follows:

$$V = \zeta(TN - C) \quad (9)$$

3.2. Maximum A Posteriori (MAP) Point Cloud Denoising

According to reference [13], the problem of estimating point cloud blocks, as shown in Equation (10), is derived as follows:

$$v_{map}(V) = \underset{V}{argmax} f(v|V)g(V) \quad (10)$$

where $f(v|V)$ represents the likelihood function and $g(V)$ is the prior probability distribution of V .

Point cloud blocks are derived from observations, making $f(v|V)$ and $f(N|U)$ equivalent. Consequently, the likelihood function $f(v|V)$ [13], formulated based on the noise model, is employed in Equations (7)–(9), as depicted in Equation (11):

$$f(v|V) = f(N|U) = \exp\{-\|N - U\|_F^2\} \quad (11)$$

According to reference [13], $g(V)$ is obtained, as shown in Equation (12):

$$g(V) = \exp\{-\beta \text{tr}(V^T L(M) V)\} \quad (12)$$

where $\beta = \frac{1}{2\sigma^2}$ represents parameters related to noise intensity, and M [13] is the Mahalanobis distance matrix between two points.

An overall Equation (13) for point cloud denoising can be obtained by combining Equations (9)–(12):

$$\underset{U, M}{argmax} \exp\{-\|U - N\|_F^2 - \|N_i - n_i\|_F^2 - \beta \text{tr}((\zeta(TN - C)^T L(M) \zeta(TN - C)) \quad (13)$$

Taking the logarithm of Equation (13) and multiplying by -1 :

$$\underset{U, M}{argmin} \|U - N\|_F^2 + \|N_i - n_i\|_F^2 + \beta \text{tr}((\zeta(TN - C)^T L(M) \zeta(TN - C)) \quad (14)$$

For any point p_i in point cloud $\mathcal{N}$, let S_i represent the average distance to k points in its neighborhood. The distance threshold [31] is obtained as shown in Equation (15).

$$d_{threshold} = \frac{\sum_{i=1}^n S_i}{N} \pm std \frac{\sum_{i=1}^n \left(S_i - \frac{\sum_{i=1}^n S_i}{N}\right)^2}{N} \quad (15)$$

The threshold of standard deviation multiples is represented by the number std . The distance between point p_i and its neighborhood, which is outside of $(\frac{\sum_{i=1}^n S_i}{N} - std \cdot \sum_{i=1}^n (S_i - \frac{\sum_{i=1}^n S_i}{N})^2 / N$ and $\frac{\sum_{i=1}^n S_i}{N} + std \cdot \sum_{i=1}^n (S_i - \frac{\sum_{i=1}^n S_i}{N})^2 / N$, is considered a Laplace noise point. Therefore, the location matrix 3 of the Laplace noise points is determined by statistical methods.

3.3. Point Cloud Noise Removal Solution

The problem in Equation (14) is solved by optimizing the point cloud U , the point cloud normal vector N_i , and the Mahalanobis distance matrix M between the two points.

Optimizing point cloud U and point cloud normal vector N_i : Initialize the Mahalanobis distance matrix M with the identity matrix I , and fix the combination Laplacian matrix L in Equation (14). Taking the derivative of the point cloud U and the normal vector N_i of the point cloud in the X direction,

$$(\beta T^T L T + I) U_x + (\beta T^T L T + I) N_{ix} = N_x + n_{ix} + \beta T^T L C_x \quad (16)$$

Taking the derivative of the point cloud $\mathbf{U}$ and the normal vector $\mathbf{N}_i$ of the point cloud in the Y direction,

$$(\beta^T \mathbf{L} \mathbf{T} + \mathbf{I}) \mathbf{U}_y + (\beta^T \mathbf{L} \mathbf{T} + \mathbf{I}) \mathbf{N}_{iy} = \mathbf{N}_y + \mathbf{n}_{iy} + \beta^T \mathbf{L} \mathbf{C}_y \quad (17)$$

Taking the derivative of the point cloud $\mathbf{U}$ and the normal vector $\mathbf{N}_i$ of the point cloud in the Z direction,

$$(\beta^T \mathbf{L} \mathbf{T} + \mathbf{I}) \mathbf{U}_z + (\beta^T \mathbf{L} \mathbf{T} + \mathbf{I}) \mathbf{N}_{iz} = \mathbf{N}_z + \mathbf{n}_{iz} + \beta^T \mathbf{L} \mathbf{C}_z \quad (18)$$

Equations (15)–(17) form a system of linear equation. These are solved iteratively using the conjugate gradient algorithm. Subsequently, the derived $\mathbf{U}$ solution is used to update $\mathbf{M}$ in the following iterations.

4. Algorithm Design

The solution process based on the point cloud denoising model, using feature maps with point normal constraints to learn denoising as shown in Algorithm 1.

Algorithm 1: Denoising for 3D point cloud based on PNCFGL

Input : $\mathbf{N}, \mathbf{n}, \mathbf{V}, \mathbf{C}, \zeta$.

Output : Denoised cloud $\mathbf{U}$.

1 : Initializing $\mathbf{U}$ with $\mathbf{N}$;

2 : for $iter = 1, 2, \dots, r$ do;

3 : Sampling s points from $\mathbf{U}$ as a block center;

4 : Find the k nearest neighbors of the center of each block to form a surface block;

5 : Optimizing objective function : $\argmin_{\mathbf{O}, \mathbf{M}} \|\mathbf{U} - \mathbf{N}\|_F^2 + \|\mathbf{N}_i - \mathbf{n}_i\|_F^2 +$

$\beta \text{tr}((\zeta(\mathbf{T}\mathbf{N} - \mathbf{C}))^T \mathbf{L}(\mathbf{M})\zeta(\mathbf{T}\mathbf{N} - \mathbf{C})), \text{ subject to : } \mathbf{N} = \mathbf{U} + \mathbf{G} + \mathbf{L};$

6 : $\mathbf{L} \leftarrow d_{\text{threshold}} = \frac{\sum_{i=1}^n S_i}{N} \pm \text{std} \cdot \frac{\sum_{i=1}^n \left(S_i - \frac{\sum_{i=1}^n S_i}{N} \right)^2}{N};$

7 : $\zeta \leftarrow f(c_i)$

8 : $\mathbf{U}$ converges;

9 : **end for**

5. Performance Evaluation

Performance evaluation of denoising algorithms can be quantitatively assessed through signal-to-noise ratio (SNR) [12] and mean squared error (MSE) [26].

The signal-to-noise ratio (SNR) is defined as shown in Equation (19):

$$\text{SNR}(\omega_1, \omega_2) 10 \lg \frac{\frac{2}{N_2} \sum_{u_i \in \omega_2} \|\mathbf{u}_j\|_2^2}{\frac{1}{N_1} \sum_{u_i \in \omega_1} \min_{u_j \in \omega_2} \|\mathbf{u}_i - \mathbf{u}_j\|_2^2 + \frac{1}{N_2} \sum_{u_i \in \omega_2} \min_{u_j \in \omega_1} \|\mathbf{u}_i - \mathbf{u}_j\|_2^2} \quad (19)$$

The mean squared error (MSE) is defined as shown in Equation (20):

$$\text{MSE}(\omega_1, \omega_2) \frac{1}{2} \left(\frac{1}{N_1} \sum_{u_i \in \omega_1} \min_{u_j \in \omega_2} \|\mathbf{u}_i - \mathbf{u}_j\|_2^2 + \frac{1}{N_2} \sum_{u_i \in \omega_2} \min_{u_j \in \omega_1} \|\mathbf{u}_i - \mathbf{u}_j\|_2^2 \right) \quad (20)$$

where $\omega_1 = \{\mathbf{u}_i\}_{i=1}^{N_1}, \mathbf{u}_i \in \mathbb{R}^3, \omega_2 = \{\mathbf{u}_j\}_{j=1}^{N_2}, \mathbf{u}_j \in \mathbb{R}^3$.

The larger the SNR, the lower the distortion of point cloud denoising and the better the denoising effect. The smaller the MSE, the lower the deviation between the ground truth value and the denoised point cloud and the better the denoising effect.

6. Results and Discussion

6.1. Comparison and Analysis of Noise Reduction Performance

To verify the effectiveness of the proposed algorithm, the common point cloud model designed by Anchor and Gargoyle, which has high-density features, was used for denoising research. Therefore, density partitioning in the proposed algorithm can be effectively performed. One may then compare the performance with existing algorithms under different noise intensity conditions.

The ground truth point cloud model for the anchor comprises 55,799 data points, and for the gargoyle, it contains 58,611 data points, as depicted in Figure 3. Gaussian and Laplace noise, with means of 0 and standard deviations of 0.02, 0.03, 0.04, and 0.05, are introduced to the ground truth point cloud model. The cases where $\sigma = 0.02$ and $\sigma = 0.04$ are illustrated in Figure 4. To facilitate the illustration of the effects post-addition and denoising, color information is incorporated into the generated model, with the color bar denoting depth information. The feature map learning algorithm, integrating point normal constraints as proposed in this paper, is employed for the denoising of 3D point cloud models across various noise intensities, the outcomes are presented in Figures 5 and 6. From the denoising results, it is evident that the algorithm proposed herein effectively eliminates Gaussian and Laplace noise from the public point cloud while appropriately preserving their original geometric features. Consequently, it constitutes an efficacious point cloud denoising algorithm.

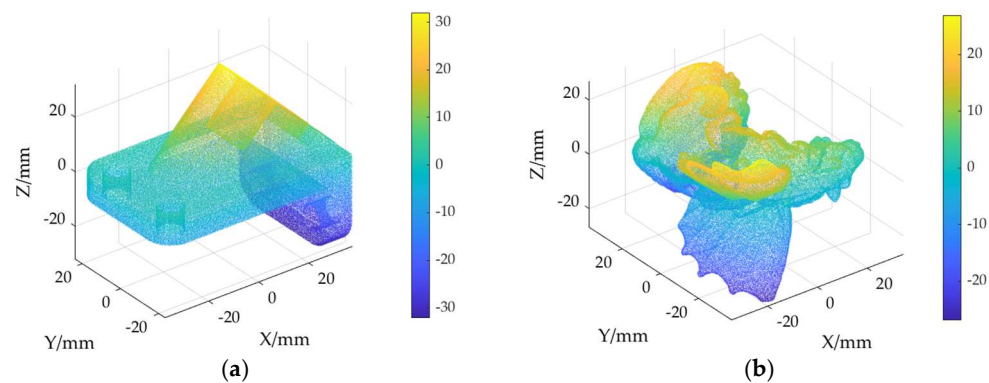


Figure 3. Point cloud model: (a) Anchor ground truth; (b) Gargoyle ground truth.

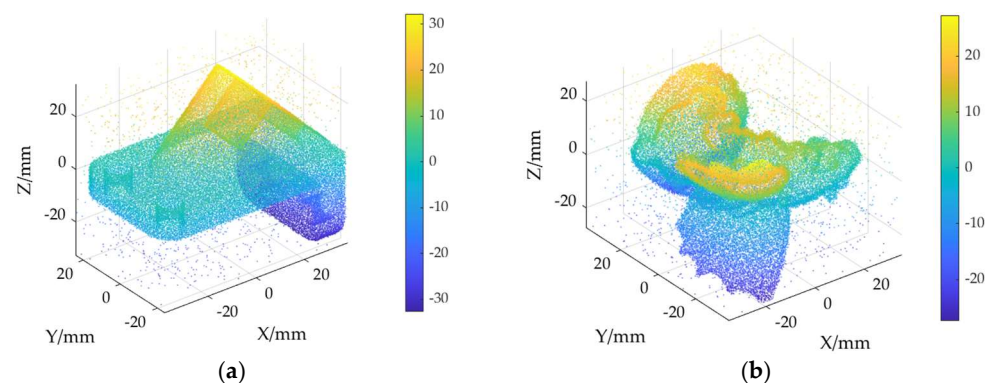


Figure 4. Point cloud model at different noise strengths: (a) Noise ($\sigma = 0.02$); (b) Noise ($\sigma = 0.04$).

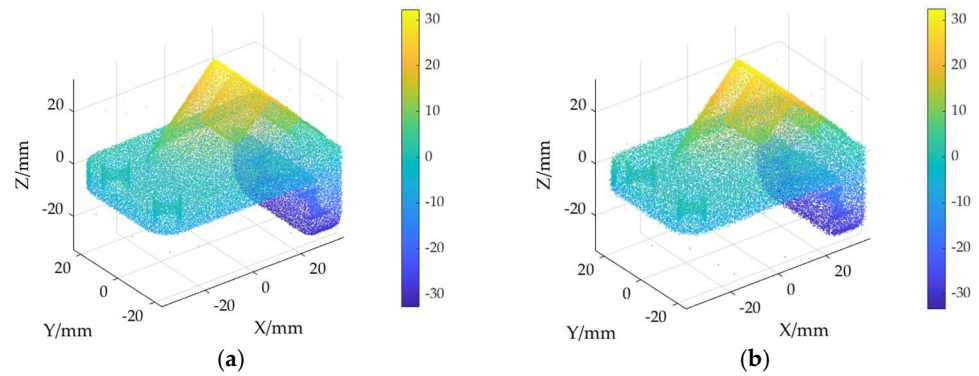


Figure 5. Noise reduction effects of PNCFGL at different strengths in Anchor model: (a) PNCFGL ($\sigma = 0.02$); (b) PNCFGL ($\sigma = 0.04$).

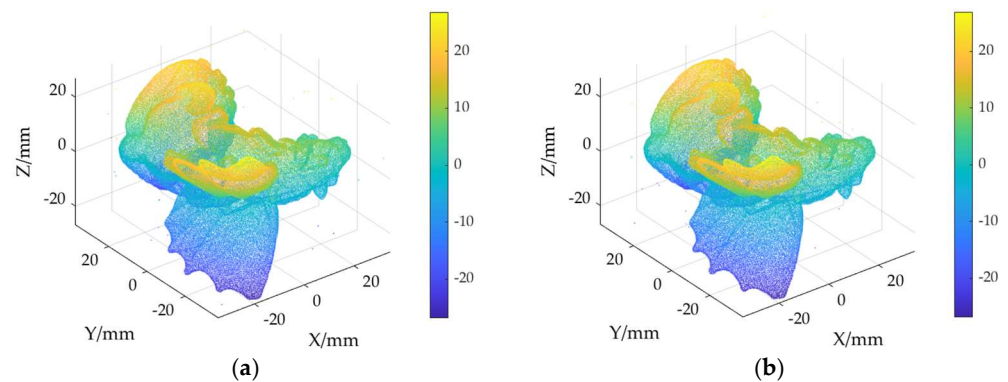


Figure 6. Noise reduction effect of PNCFGL at different strengths in Gargoyle model: (a) PNCFGL ($\sigma = 0.02$); (b) PNCFGL ($\sigma = 0.04$).

The noisy point cloud models at varying intensities are presented in Figure 4, denoised individually by APSS, NLD and MRPCA, with corresponding results depicted in Figures 7–12. In Figures 7 and 8, uneven color distribution and blurred planar features are observed on the surface of the denoised point cloud model by the APSS algorithm. Additionally, the increased presence of outlier noise is noted. Comparatively, Figures 9 and 10, as well as Figures 11 and 12, display relatively uniform color distribution on the surfaces. However, some noise persists away from the main body. The analysis of the results in Figures 7–12 indicates inadequate elimination of Laplace noise. Subsequently, the PNCFGL algorithm was used for noise removal, and the results showed that it significantly reduced abnormal noise and improved the accuracy of local features.

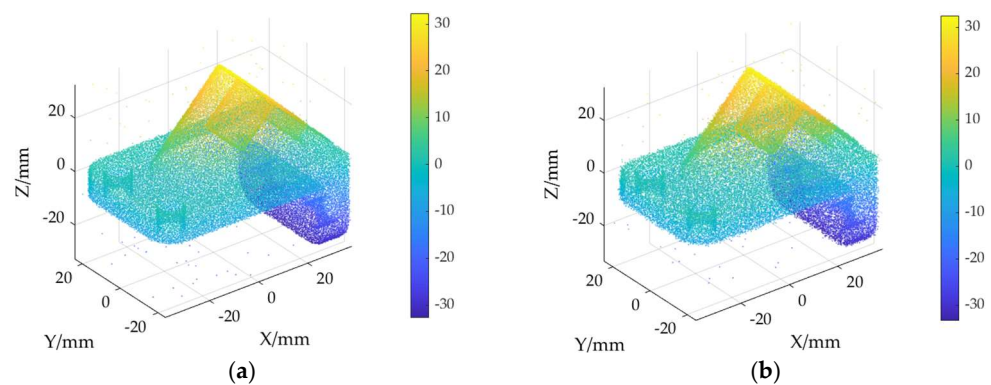


Figure 7. Noise reduction effect of APSS at different noise strengths in Anchor model: (a) APSS ($\sigma = 0.02$); (b) APSS ($\sigma = 0.04$).

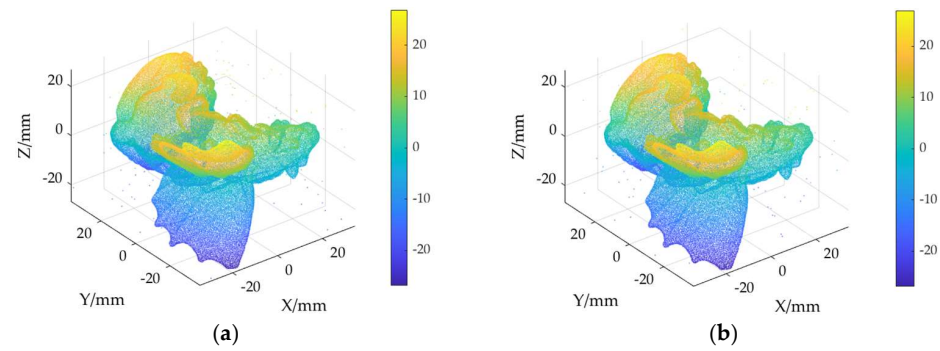


Figure 8. Noise reduction effect of APSS at different noise intensities in Gargoyles model: (a) APSS ($\sigma = 0.02$); (b) APSS ($\sigma = 0.04$).

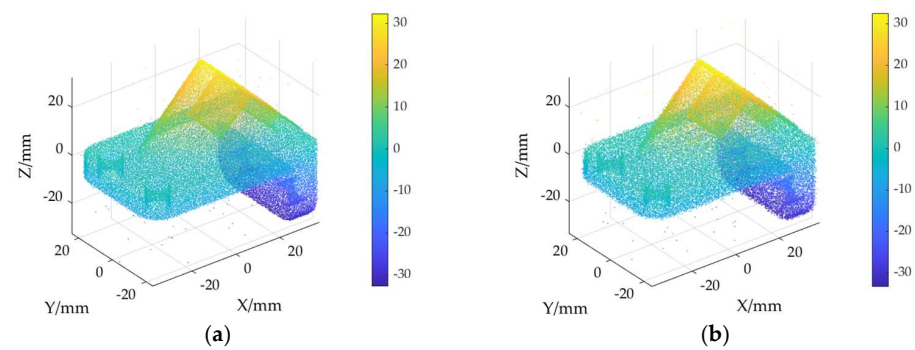


Figure 9. Noise reduction effects of NLD at different strengths in Anchor model: (a) NLD ($\sigma = 0.02$); (b) NLD ($\sigma = 0.04$).

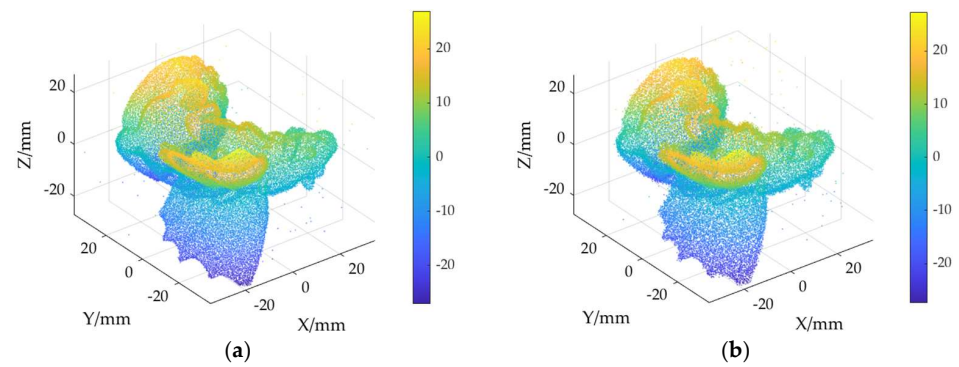


Figure 10. Noise reduction effects of NLD at different strengths in Gargoyles model: (a) NLD ($\sigma = 0.02$); (b) nld ($\sigma = 0.04$).

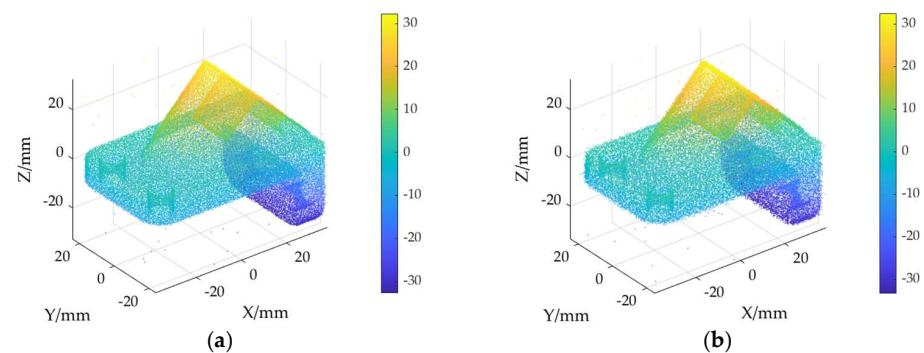


Figure 11. Noise removal effect of MRPCA at different strengths in Anchor model: (a) MRPCA ($\sigma = 0.02$); (b) MRPCA ($\sigma = 0.04$).

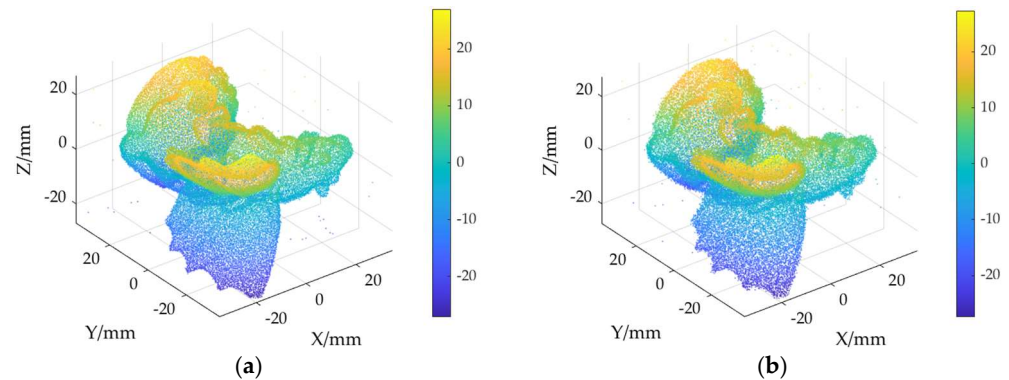


Figure 12. De-noising effects of MRPCA at different strengths in Gargoyle model: (a) MRPCA ($\sigma = 0.02$); (b) MRPCA ($\sigma = 0.04$).

As depicted in Table 1, under the MSE metrics, an anchor-based point cloud model after denoising at four noise levels with different values of σ (0.02–0.05), at $\sigma = 0.05$, the anchor is 0.267 for APSS, 0.331 for NLD, 0.315 for AWLOP, and 0.253 for MRPCA and 0.245 for the algorithm in this paper, which is a reduction of 0.022 compared to APSS, 0.086 compared to NLD, 0.070 compared to AWLOP, and 0.008 compared to MRPCA. After denoising, the average MSE of APSS, AWLOP, NLD, and MRPCA are 0.242, 0.280, 0.281, and 0.232, respectively, and the average MSE of the algorithm in this paper is 0.227, which is reduced by 0.015 compared to APSS, 0.054 compared to NLD, 0.053 compared to AWLOP, and 0.005 compared to MRPCA. Quantitative analysis underscores the superior denoising efficacy of this paper's algorithm, characterized by lower deviation between the ground truth value and the denoised point cloud.

Table 1. MSE of five different denoising algorithms for the anchor model.

Noise Levels	$\sigma = 0.02$	$\sigma = 0.03$	$\sigma = 0.04$	$\sigma = 0.05$	Average	Time
Noisy	0.259	0.322	0.372	0.417	0.343	754 s
APSS	0.208	0.239	0.254	0.267	0.242	891 s
AWLOP	0.237	0.259	0.306	0.315	0.280	780 s
NLD	0.231	0.265	0.297	0.331	0.281	718 s
MRPCA	0.202	0.230	0.242	0.253	0.232	868 s
RSLDM	0.201	0.228	0.240	0.249	0.230	1073 s
Ours	0.199	0.226	0.237	0.245	0.227	782 s

As depicted in Table 2, under the MSE metrics, an anchor-based point cloud model after denoising at four noise levels with different values of σ (0.02–0.05), at $\sigma = 0.05$, the anchor is 47.09 DB for APSS, 44.94 DB for NLD, 45.44 DB for AWLOP, 47.64 DB for MRPCA and 47.81 DB for the algorithm in this paper, which is an improvement of 0.72 DB over APSS, 2.87 DB over NLD, 2.37 DB over AWLOP, and 0.17 DB over MRPCA. After denoising, the average SNR of APSS, NLD, AWLOP, and MRPCA are 48.14 DB, 46.66 DB, 46.54 DB, and 48.55 DB, respectively. The average SNR of the algorithm in this paper is 48.68 DB, which is 0.54 DB better than APSS, 2.14 DB better than AWLOP, 2.02 DB better than NLD, and 0.13 DB better than MRPCA. It is quantitatively analyzed to be better than the first three algorithms, with the lowest distortion in the denoising process.

In terms of the comparison of noise point clouds under different intensities with the latest graph learning algorithm RSLDM, according to Tables 1 and 2, the average MSE of RSLDM is 0.230, while the algorithm in this paper is 0.227, which is an improvement of 0.003. The average SNR of RSLDM is 48.62 DB, while the algorithm proposed in this paper is 48.68 DB, which is an improvement of 0.06 DB. However, this algorithm has improved computational efficiency by 27% compared to RSLDM due to RSLDM dividing point clouds into blocks based on the k-neighborhood, while our algorithm divides point clouds into blocks based on density. Density partitioning utilizes the distribution density

of points in the point cloud to determine the partitioning boundary, without the need to calculate the k-neighborhood of each point, reducing computational complexity and time consumption. Furthermore, it can determine the partition boundary based on the actual density distribution of point cloud data, which performs better when dealing with point clouds with uneven distribution or large-scale density changes. The impact of density partitioning on noise is relatively small. It determines the boundary through the overall density distribution, rather than relying solely on the local neighborhood of each point.

Table 2. SNR of five different denoising algorithms for the anchor model.

Noise Level	$\sigma = 0.02$	$\sigma = 0.03$	$\sigma = 0.04$	$\sigma = 0.05$	Average	Time
Noisy	47.41	45.25	43.18	42.65	44.62	754 s
APSS	49.61	48.24	47.60	47.09	48.14	891 s
NLD	48.53	47.16	46.02	44.94	46.66	780 s
AWLOP	48.31	46.69	45.74	45.44	46.54	718 s
MRPCA	49.88	48.60	48.09	47.64	48.55	868 s
RSLDM	49.97	48.65	48.12	47.76	48.62	1073 s
Ours	50.04	48.70	48.15	47.81	48.68	782 s

The algorithm in this article combines point cloud density information and the consistency of normal vector direction to simultaneously remove Gaussian noise and Laplacian noise. Density information can help identify noise points in sparse regions, while normal vector information can help identify noise points with inconsistent normal vectors. Combining these two types of information can effectively combat different types of noise and improve the accuracy and reliability of point cloud data.

6.2. Noise Reduction Applications of 3D Point Clouds

To further validate the proposed PNCFGL model, a Lidar scanner was used to capture point clouds of four objects, as shown in Figure 13. The four objects were pipe joints, pipe rings, rear brackets, and front brackets, as shown in Figure 14. After denoising, the MSE and SNR results are shown in Tables 3 and 4. The environment of the denoising equipment used was as follows: the CPU model was AMD Ryzen7 3800X (Santa Clara, CA, USA), with a basic frequency of 3.90 GHz; the graphics card model was RTX3090 (Nvidia Corporation, Santa Clara, CA, USA); and the memory was 32 GB DDR4.

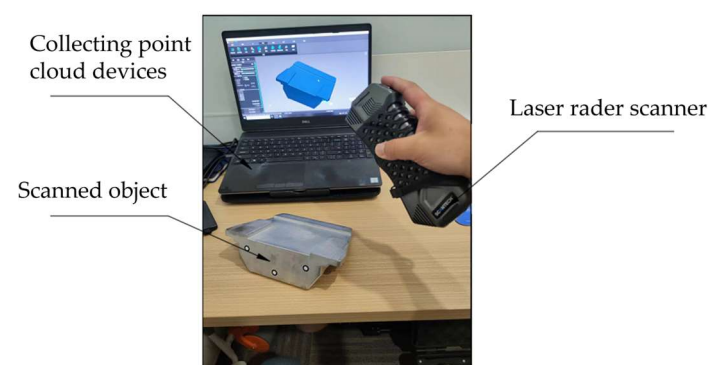


Figure 13. Laser radar scanner.

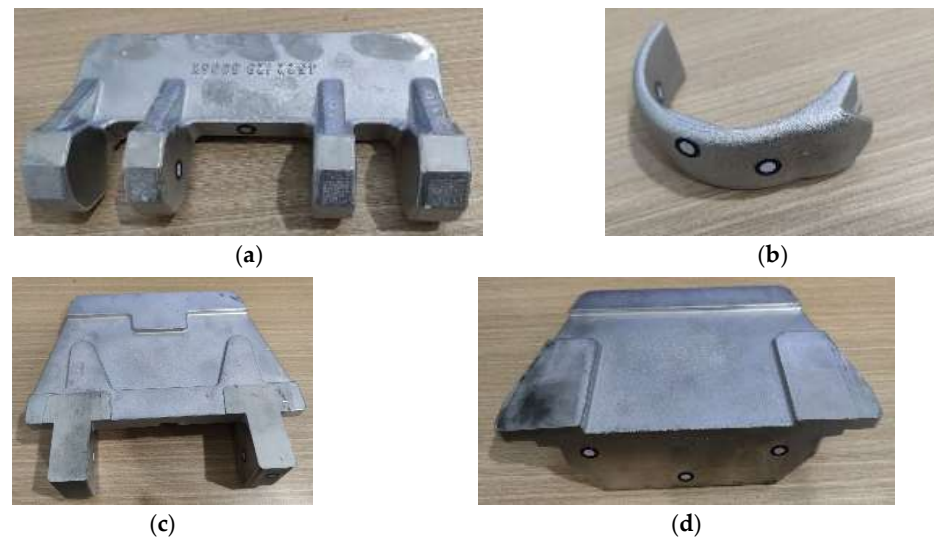


Figure 14. Scanning objects: (a) pipe fitting; (b) pipe circle; (c) rear support; (d) front support.

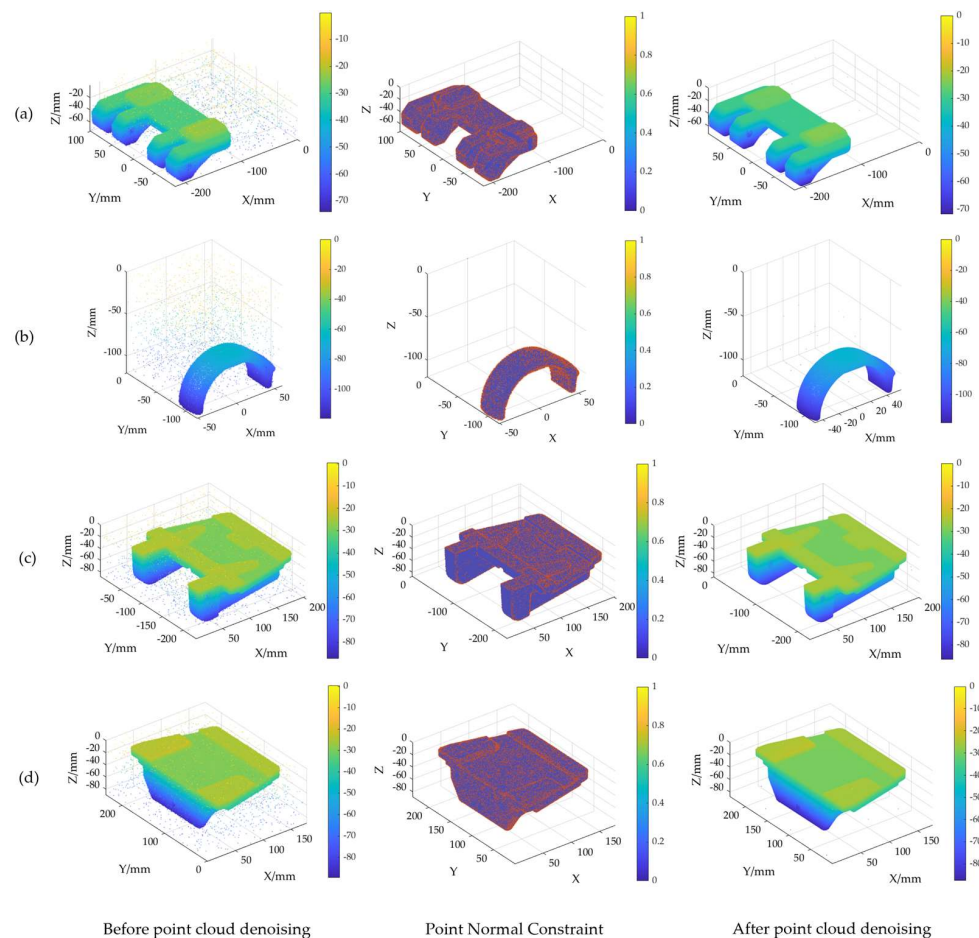
Table 3. Comparison of MSE for objects a–d under different algorithms.

Method	Noisy	APSS	NLD	AWLOP	MRPCA	RSLDM	Ours
$\sigma = 0.02$							
a	0.103	0.052	0.075	0.081	0.047	0.046	0.043
b	0.116	0.065	0.083	0.092	0.057	0.057	0.054
c	0.119	0.061	0.080	0.089	0.056	0.054	0.052
d	0.113	0.063	0.082	0.091	0.057	0.056	0.053
Average	0.112	0.060	0.080	0.088	0.054	0.053	0.505
$\sigma = 0.03$							
a	0.160	0.077	0.102	0.092	0.065	0.064	0.062
b	0.159	0.075	0.104	0.096	0.068	0.063	0.063
c	0.161	0.079	0.103	0.098	0.069	0.066	0.064
d	0.160	0.076	0.102	0.091	0.065	0.063	0.062
Average	0.160	0.076	0.103	0.094	0.067	0.064	0.063
$\sigma = 0.04$							
a	0.207	0.089	0.132	0.141	0.077	0.075	0.072
b	0.195	0.078	0.121	0.130	0.064	0.064	0.062
c	0.204	0.090	0.130	0.142	0.078	0.077	0.073
d	0.203	0.088	0.127	0.140	0.075	0.073	0.071
Average	0.202	0.086	0.128	0.138	0.074	0.072	0.070
$\sigma = 0.05$							
a	0.253	0.103	0.167	0.151	0.089	0.087	0.081
b	0.242	0.096	0.158	0.141	0.082	0.076	0.073
c	0.254	0.102	0.168	0.153	0.091	0.086	0.082
d	0.257	0.108	0.170	0.156	0.094	0.088	0.084
Average	0.252	0.102	0.166	0.150	0.089	0.084	0.080

Gaussian noise and Laplacian noise, each with mean values of 0 and standard deviations of 0.02, 0.03, 0.04, and 0.05, were introduced into the scanned point cloud. The denoising performance of PNCFGl was then evaluated using mean squared error (MSE) and signal-to-noise ratio (SNR), as illustrated in Figure 15. Post-PNCFGl denoising, the outlier noise associated with the four objects witnessed a notable reduction, while preserving the local feature structure effectively.

Table 4. Comparison of SNR for objects a–d under different algorithms.

Method	Noisy	APSS	NLD	AWLOP	MRPCA	RSLDM	Ours
$\sigma = 0.02$							
a	81.55	83.75	82.65	82.45	84.02	84.10	84.18
b	76.32	78.30	77.32	76.69	78.69	78.75	78.83
c	79.50	81.63	80.25	80.33	82.01	82.10	82.16
d	78.68	80.88	79.90	79.58	81.12	81.25	81.31
Average	79.01	81.14	80.03	79.76	81.46	81.55	81.58
$\sigma = 0.03$							
a	77.08	80.07	78.99	78.50	80.43	80.44	80.53
b	74.64	77.63	76.55	76.16	78.06	78.12	78.19
c	77.34	80.23	79.09	78.68	80.56	80.61	80.69
d	76.56	79.39	78.30	77.89	79.78	79.87	79.91
Average	76.40	79.33	78.23	77.81	79.71	79.76	79.83
$\sigma = 0.04$							
a	74.02	78.56	76.90	76.62	78.99	79.01	79.05
b	71.66	76.08	74.55	74.25	76.54	76.54	76.63
c	74.17	78.66	77.09	76.86	79.18	79.20	79.24
d	73.52	77.96	76.42	76.32	78.43	78.47	78.53
Average	73.34	77.82	76.24	76.01	78.29	78.28	78.36
$\sigma = 0.05$							
a	72.52	76.96	74.81	75.31	77.51	77.56	77.68
b	70.06	74.50	72.45	72.79	75.06	75.10	75.20
c	72.65	77.19	75.12	75.54	77.86	77.87	77.91
d	72.15	76.52	74.45	74.92	77.12	77.18	77.28
Average	71.85	76.29	74.21	74.64	76.89	76.92	77.02

**Figure 15.** Comparison of objects (a–d) before and after PNCFGL noise reduction.

As depicted in Table 3, under the MSE metrics, at $\sigma = 0.02$, the object a is 0.052 for APSS, 0.075 for NLD, 0.081 for AWLOP, and 0.047 for MRPCA and 0.043 for the algorithm in this paper, which is a reduction of 0.009 compared to APSS, 0.032 compared to NLD, 0.038 compared to AWLOP, and 0.004 compared to MRPCA. At $\sigma = 0.03$, the object b is 0.075 for APSS, 0.104 for NLD, 0.096 for AWLOP, and 0.068 for MRPCA and 0.063 for the algorithm in this paper, which is a reduction of 0.012 compared to APSS, 0.041 compared to NLD, 0.033 compared to AWLOP, and 0.005 compared to MRPCA. At $\sigma = 0.04$, the object c is 0.090 for APSS, 0.130 for NLD, 0.142 for AWLOP, and 0.078 for MRPCA and 0.073 for the algorithm in this paper, which is a reduction of 0.012 compared to APSS, 0.057 compared to NLD, 0.069 compared to AWLOP and 0.005 compared to MRPCA. At $\sigma = 0.05$, the object d is 0.108 for APSS, 0.170 for NLD, 0.156 for AWLOP, and 0.094 for MRPCA and 0.084 for the algorithm in this paper, which is a reduction of 0.024 compared to APSS, 0.086 compared to NLD, 0.072 compared to AWLOP, and 0.010 compared to MRPCA. Quantitative analysis reveals that the algorithm in this study demonstrates reduced point cloud deviation and enhanced performance post-noise removal.

As depicted in Table 4, under the SNR metrics, at $\sigma = 0.02$, the object a is 83.75 DB for APSS, 82.65 DB for NLD, 82.45 DB for AWLOP, and 84.02 DB for MRPCA and 84.18 DB for the algorithm in this paper, which is an improvement of 0.43 DB over APSS, 1.53 DB over NLD, 1.73 DB over AWLOP, and 0.16 DB over MRPCA. At $\sigma = 0.03$, the object b is 76.63 DB for APSS, 76.55 DB for NLD, 76.16 DB for AWLOP, 78.06 DB for MRPCA and 78.19 DB for the algorithm in this paper, which is an improvement of 1.56 DB over APSS, 1.64 DB over NLD, 2.03 DB over AWLOP, and 0.13 DB over MRPCA. At $\sigma = 0.04$, the object c is 78.66 DB for APSS, 77.09 DB for NLD, 76.86 DB for AWLOP, 79.18 DB for MRPCA and 79.24 DB for the algorithm in this paper, which is an improvement of 0.58 DB over APSS, 2.15 DB over NLD, 2.38 DB over AWLOP, and 0.06 DB over MRPCA. At $\sigma = 0.05$, the object d is 76.52 DB for APSS, 74.45 DB for NLD, 74.92 DB for AWLOP, 77.12 DB for MRPCA and 77.28 DB for the algorithm in this paper, which is an improvement of 0.76 DB over APSS, 2.83 DB over NLD, 2.36 DB over AWLOP, and 0.16 DB over MRPCA. Quantitative analysis indicates that the denoising procedure implemented by the algorithm proposed in this paper exhibits diminished distortions and superior denoising efficacy.

7. Conclusions

In this study, a novel point normal constrained feature map learning approach is proposed for effectively mitigating mixed noise within point clouds. Specifically, the reinforcement of normal vectors with subsurface orthogonality conditions is achieved through the incorporation of Dirichlet energy and coupled orthogonality bias terms, aimed at penalizing differences among neighboring normal vectors. These imposed constraints significantly enhance the accuracy and robustness of feature map learning, thus facilitating the denoising of point clouds exhibiting prominent geometric features. Denoising efficacy is evaluated through quantitative metrics, including signal-to-noise ratio (SNR) and mean squared error (MSE), applied to the denoising of the anchor model across four distinct noise levels. Compared with APSS, AWLOP, NLD, MRPCA, and RSLDM, the proposed algorithm has an average SNR improvement of 2.14 DB and an average MSE decrease of 0.054, while achieving similar computational efficiency. While the denoising performance and RSLDM have slightly improved, the computational efficiency has increased by 27%. This algorithm achieves a balance between denoising performance and computational efficiency. Therefore, this algorithm not only effectively removes mixed noise, but also preserves and improves computational efficiency, providing a feasible solution for point cloud data processing. However, due to the need for point cloud density partitioning during its denoising process, it is currently only applicable to high-density point cloud features. Future work will focus on extending this method to low-density application scenarios, further optimizing the algorithm to meet different practical application needs, improving the real-time performance and stability of the algorithm and exploring more

constraints and regularization terms to further improve the effectiveness and accuracy of point cloud denoising.

Author Contributions: Conceptualization, Y.L.; methodology, Z.F.; software, L.S. and M.H.S.; validation, L.X. and L.S.; formal analysis, Z.F. and M.H.S.; investigation, L.S.; resources, L.X.; data curation, Y.L.; writing—original draft preparation, Z.F. and M.H.S.; writing—review and editing, Z.F.; visualization, L.S. and M.H.S.; supervision, Z.F.; project administration, Y.L.; funding acquisition, L.X. All authors have read and agreed to the published version of the manuscript.

Funding: Supported in part by the University Synergy Innovation Program of Anhui Province of China under Grant No. GXXT-2023-038 and by the Open Research Fund of Anhui Key Laboratory of Detection Technology and Energy Saving Devices under Grant No. JCKJ2022A03.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: Authors Lijin Xu and Liping Shi were employed by the company Anhui Gongchuang Industrial Robot Innovation Center Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Du, X.; Zhang, T.; Xie, H. DualShape. Sketch-based 3D Shape Design with Part Generation and Retrieval. *IEEE Access* **2024**, *12*, 18888–18900. [\[CrossRef\]](#)
2. Park, M.; Park, C.; Kwon, N.K. Autonomous Driving of Mobile Robots in Dynamic Environments Based on Deep Deterministic Policy Gradient: Reward Sha and Hindsight Experience Replay. *Biomimetics* **2024**, *9*, 51. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Pintos Gómez de las Heras, B.; Martínez-Tomás, R.; Cuadra Troncoso, J.M. Self-Learning Robot Autonomous Navigation with Deep Reinforcement Learning Techniques. *Appl. Sci.* **2024**, *14*, 366. [\[CrossRef\]](#)
4. Ping, G.; Esfahani, M.A.; Chen, J.; Wang, H. Visual enhancement of single-view 3D point cloud reconstruction. *Comput. Graph.* **2022**, *102*, 112–119. [\[CrossRef\]](#)
5. Wang, J.; Zheng, X.; Xu, Z. Quasi-Dense Matching Algorithm for Close-Range Image Combined with Feature Line Constraint. *IEEE Access* **2022**, *10*, 117914–117924. [\[CrossRef\]](#)
6. Holmström, S.T.; Baran, U.; Urey, H. MEMS laser scanners: A Review. *J. Microelectromech. Syst.* **2014**, *23*, 259–275. [\[CrossRef\]](#)
7. Cutti, A.G.; Santi, M.G.; Hansen, A.H.; Fatone, S. Accuracy, Repeatability, and Reproducibility of a Hand-Held Structured-Light 3D Scanner across Multi-Site Settings in Lower Limb Prosthetics. *Sensors* **2024**, *24*, 2350. [\[CrossRef\]](#)
8. Hu, Z.; Jiang, C.; Zhu, J.; Zhao, Y.; Liu, Z.; Yuan, Y.; Wang, Y. Efficient and robust chaos single photon Lidar. *Opt. Laser Technol.* **2024**, *174*, 110623. [\[CrossRef\]](#)
9. Gou, F.; Li, Y.; Hou, C.; Wang, H. 3D point cloud global denoising algorithm based on different noise characteristics. *Meas. Sci. Technol.* **2023**, *35*, 015032. [\[CrossRef\]](#)
10. Ping, G.; Wang, H. PushNet. 3D reconstruction from a single image by pushing. *Neural Comput. Appl.* **2024**, *36*, 6629–6641. [\[CrossRef\]](#)
11. Wang, L.; Yang, B.; Ye, H.; Cao, F. Two-view point cloud registration network: Feature and geometry. *Appl. Intell.* **2024**, *54*, 3135–3151. [\[CrossRef\]](#)
12. Zeng, J.; Cheung, G.; Ng, M.; Pang, J.; Yang, C. 3D point cloud denoising using graph Laplacian regularization of a low dimensional manifold model. *IEEE Trans. Image Process.* **2019**, *29*, 3474–3489. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Hu, W.; Gao, X.; Cheung, G.; Guo, Z. Feature graph learning for 3D point cloud denoising. *IEEE Trans. Signal Process.* **2020**, *68*, 2841–2856. [\[CrossRef\]](#)
14. Sun, Y.; Schaefer, S.; Wang, W. Denoising point sets via L0 minimization. *Comput. Aided Geom. Des.* **2015**, *35*, 2–15. [\[CrossRef\]](#)
15. Avron, H.; Sharf, A.; Greif, C.; Cohen-Or, D. ℓ_1 -sparse reconstruction of sharp point set surfaces. *ACM Trans. Graph. (TOG)* **2010**, *29*, 1–12. [\[CrossRef\]](#)
16. Mattei, E.; Castrodad, A. Point cloud denoising via moving RPCA. *Comput. Graph. Forum* **2017**, *36*, 123–137. [\[CrossRef\]](#)
17. Buades, A.; Coll, B.; Morel, J.M. A non-local algorithm for image denoising. *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* **2005**, *2*, 60–65.
18. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process.* **2007**, *16*, 2080–2095. [\[CrossRef\]](#)
19. Deschaut, J.E.; Goulette, F. Point cloud non local denoising using local surface descriptor similarity. *PCV Photogramm. Comput. Vision.* **2010**, *120*, 108128. [\[CrossRef\]](#)

20. Alexa, M.; Behr, J.; Cohen-Or, D.; Fleishman, S.; Levin, D.; Silva, C.T. Computing and rendering point set surfaces. *IEEE Trans. Vis. Comput. Graph.* **2003**, *9*, 3–15. [[CrossRef](#)]
21. Guennebaud, G.; Gross, M. Algebraic point set surfaces. In Proceedings of the SIGGRAPH07: Special Interest Group on Computer Graphics and Interactive Techniques Conference, San Diego, CA, USA, 5–9 August 2007.
22. Huang, H.; Li, D.; Zhang, H.; Ascher, U.; Cohen-Or, D. Consolidation of unorganized point clouds for surface reconstruction. *ACM Trans. Graph. (TOG)* **2009**, *28*, 1–7. [[CrossRef](#)]
23. Huang, H.; Wu, S.; Gong, M.; Cohen-Or, D.; Ascher, U.; Zhang, H. Edge-aware point set resampling. *ACM Trans. Graph. (TOG)* **2013**, *32*, 1–12. [[CrossRef](#)]
24. Dinesh, C.; Cheung, G.; Bajić, I.V. Point cloud denoising via feature graph laplacian regularization. *IEEE Trans. Image Process.* **2020**, *29*, 4143–4158. [[CrossRef](#)] [[PubMed](#)]
25. Schoenenberger, Y.; Paratte, J.; Vandergheynst, P. Graph-based denoising for time-varying point clouds. In Proceedings of the 3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON), Lisbon, Portugal, 8–10 July 2015; IEEE: New York, NY, USA, 2015; pp. 1–4.
26. Zhang, X.; Cheung, G.; Pang, J.; Sanghvi, Y.; Gnanasambandam, A.; Chan, S.H. Graph-based depth denoising & dequantization for point cloud enhancement. *IEEE Trans. Image Process.* **2022**, *31*, 6863–6878. [[CrossRef](#)] [[PubMed](#)]
27. Liu, Y.; Zou, B.; Xu, J.; Yang, S.; Li, Y. Denoising for 3D point cloud based on regularization of a statistical low-dimensional manifold. *Sensors* **2022**, *22*, 2666. [[CrossRef](#)]
28. Han, X.F.; Jin, J.S.; Wang, M.J.; Jiang, W.; Gao, L.; Gao, L. A review of algorithms for filtering the 3D point cloud. *Signal Process. Image Commun.* **2017**, *57*, 103–112. [[CrossRef](#)]
29. Shrestha, A.; Mahmood, A. Review of deep learning algorithms and architectures. *IEEE Access* **2019**, *7*, 53040–53065. [[CrossRef](#)]
30. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
31. Rakotosaona, M.J.; La Barbera, V.; Guerrero, P.; Mitra, N.J.; Ovsjanikov, M. Pointcleannet: Learning to denoise and remove outliers from dense point clouds. *Comput. Graph. Forum* **2020**, *39*, 185–203. [[CrossRef](#)]
32. Xiao, D.; Shi, Z.; Li, S.; Deng, B.; Wang, B. Point normal orientation and surface reconstruction by incorporating isovalue constraints to poisson equation. *Comput. Aided Geom. Des.* **2023**, *103*, 102195. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.