

Generated Realistic Noise and Rotation-Equivariant Models for Data-Driven Mesh Denoising

Sipeng, Yang^a, Wenhui, Ren^a, Xiwen, Zeng^a, Qingchuan, Zhu^a, Hongbo, Fu^b, Kaijun, Fan^c, Lei, Yang^c, Jingping, Yu^c, Qilong, Kou^c and Xiaogang, Jin^{a,*}

^aState Key Lab of CAD&CG, Zhejiang University, China

^bSchool of Creative Media, City University of Hong Kong, Hong Kong

^cLightSpeed Studios, Tencent, China

ARTICLE INFO

Keywords:

Mesh denoising

Realistic noise generation

Equivariant models

ABSTRACT

3D mesh denoising is a crucial pre-processing step in many graphics applications. However, existing data-driven mesh denoising models, primarily trained on synthetic white noise, are less effective when applied to real-world meshes with the noise of complex intensities and distributions. Moreover, how to comprehensively capture information from input meshes and apply suitable denoising models for feature-preserving mesh denoising remains a critical and unresolved challenge. This paper presents a rotation-Equivariant model-based Mesh Denoising (EMD) model and a Realistic Mesh Noise Generation (RMNG) model to address these issues. Our EMD model leverages rotation-equivariant features and self-attention weights of geodesic patches for more effective feature extraction, thereby achieving SOTA denoising results. The RMNG model, based on the Generative Adversarial Networks (GANs) architecture, generates massive amounts of realistic noisy and noiseless mesh pairs data for data-driven mesh denoising model training, significantly benefiting real-world denoising tasks. To address the smooth degradation and loss of sharp edges commonly observed in captured meshes, we further introduce varying levels of Laplacian smoothing to input meshes during the paired training data generation, endowing the trained denoising model with feature recovery capabilities. Experimental results demonstrate the superior performance of our proposed method in preserving fine-grained features while removing noise on real-world captured meshes.

1. Introduction

3D scanning and photogrammetry are prevalent approaches for capturing 3D models from the real world and are extensively utilized in various applications such as cultural heritage, video game asset production, and reverse engineering. Recent sensor hardware advancements have significantly improved the precision and complexity of captured 3D meshes. However, the noise over captured models has not subsided with the enhanced sensor precision. Instead, detecting and removing the noise present in the feature-rich meshes becomes increasingly difficult, posing greater challenges for mesh denoising algorithms.

Mesh denoising aims to recover a clean mesh from its noisy version. Early techniques (Desbrun et al. (1999); Lee and Wang (2005)) often rely on assumptions about noise or priors for noiseless meshes, necessitating careful parameter tuning and often failing in feature preservation (Wang et al. (2016)). Recent trends have shifted towards data-driven methods that learn parameters directly from datasets. These approaches (Zhao et al. (2022); Shen et al. (2022)), leveraging well-designed representations of 3D meshes and the powerful modeling capabilities of deep learning, have demonstrated effectiveness in noise removal and feature preservation.

However, our observations reveal that existing data-driven approaches have limitations in dealing with irregular meshes, impacting their overall performance. For instance, patch-based representation methods (Li et al. (2020); Shen et al. (2022)), which combine features of adjacent faces for each facet or represent a mesh as a graph (Armando et al. (2020)), neglect the state information of facets on a surface. These approaches do not take into account information regarding the areas of facets and the distances between facets, thus limiting the learning efficacy of normal prediction models. Another strategy involves resampling mesh patches to transform irregular 3D meshes into regular features,

*Corresponding author

 jin@cad.zju.edu.cn (X. Jin)

 <http://www.cad.zju.edu.cn/home/jin/> (X. Jin)

thereby preserving the state information of the facets. Methods proposed by Wei et al. (2018) and Wang et al. (2019) utilize sampled nonlocal patch matrices but do not fully exploit the relative spatial positions information among samples. A recent work, local surface descriptors (LSD, Zhao et al. (2022)), addresses these issues by applying convolutional neural networks (CNNs) to the locally sampled surface of each face. However, due to the arbitrary orientation of surface sampling, the LSD method is susceptible to the rotation ambiguity problem (Wiersma et al. (2020)), which hampers its capability to achieve optimal results.

Besides the learning paradigms, large-scale training data also plays a significant role in the performance of data-driven denoising methods. However, collecting massive real noisy-clean datasets is generally expensive and even inaccessible for some tasks. A seemingly effective alternative is using manual additive white noise (AWN) to degrade noiseless meshes, thereby obtaining noisy-clean paired training data (Wang et al. (2016)). But since real noise patterns are much more complex, denoising models trained on AWN datasets usually exhibit inferior performance in real-world tasks (Wu et al. (2020)).

In this paper, we present a rotation-equivariant model based mesh denoising (*EMD*) method for removing mesh noise, as well as a realistic mesh noise generation (*RMNG*) model to generate substantial amounts of noisy-clean data for training the denoising model. For the denoising model *EMD*, our core idea is to perform convolution and pooling layers on meshes to extract features at various scales, followed by predicting noiseless facet normals. First, we employ geodesic patches (Masci et al. (2015)) to aggregate features of a facet and its neighbors, ensuring the collection of comprehensive area and relative position information of the facets. Then, to avoid the rotation ambiguity problem encountered with vanilla CNNs, an equivariant learning model is required to handle the geodesic patches with a rotational degree of freedom. To this end, we introduce an efficient equivariant model for surface learning, utilizing equivariant steerable CNNs (Weiler and Cesa (2019)) in Euclidean spaces \mathbb{R}^2 . Besides, we design a self-attention reweighting module for the extracted features to enhance edge preservation performance. Finally, the predicted noiseless facet normals are refined and then used to update vertex positions for removing noise as previous works (Sun et al. (2007)).

For generating noisy-clean training data, we propose the *RMNG* model to learn realistic noise patterns from given noisy meshes. The learned *RMNG* model is then used to degrade noiseless meshes, thereby producing extensive pairs of noisy-clean data. Compared to methods for noise generation on images (Cai et al. (2021); Lee and Kim (2022)), generating realistic noise on irregular meshes faces additional challenges, such as maintaining consistency between vertex positions and noise signals, preserving the underlying shape, smooth degradation, and so on. To handle these concerns, we propose generating corresponding zero-mean noise on noiseless meshes using rotation-equivariant models based on generative adversarial networks (GANs). The proposed model uses sampled vertex maps from a noiseless mesh to create corresponding noisy maps. Random noise is added to the early layers of a generator to create variable noise patterns. Then, a discriminator providing rotation-invariant judgments is used to differentiate between real and generated noisy samples. We introduce two additional loss functions to ensure the consistency of underlying shapes and the variety of noise patterns. Furthermore, when creating the noisy-clean training dataset, we apply two levels of Laplacian smoothing to the input mesh to give the denoising model sharp feature recovery abilities and thus reducing the smooth degradation issue (Nocerino et al. (2020)).

In summary, our method effectively exploits the benefits of a data-driven strategy. Requiring only a few noisy meshes from a user's project, our method learns inherent noise patterns to generate a large-scale and customized training dataset using existing noiseless meshes to train denoising models. This approach provides a practical and user-friendly solution readily adaptable to various real-world denoising tasks. Our paper makes the following contributions:

- We introduce an equivariant model-based mesh denoising (*EMD*) method to achieve SOTA results. The method can better preserve geometric details by utilizing rotation-equivariant features and self-attention weights.
- We provide a realistic mesh noise generation (*RMNG*) model to produce realistic noisy-clean pairs for denoising methods, greatly benefiting real-world denoising tasks.
- We propose incorporating smoothing degradation into the noisy-clean data generation process, which provides feature recovery capabilities to the trained mesh denoising model.

2. Related Work

2.1. Mesh Denoising Methods

Early mesh denoising methods (Desbrun et al. (1999)) perform Laplacian smoothing or its variants on vertices to remove high-frequency noise, often resulting in the smoothing of sharp features. To mitigate the loss of sharp features,

subsequent studies propose to use feature-preserving filters for mesh denoising, such as anisotropic diffusion (Clarenz et al. (2000)) and bilateral filtering (Lee and Wang (2005)), which necessitate careful parameter selection. More recent filter-based approaches (Sun et al. (2007); Zheng et al. (2010); Wei et al. (2014); Lu et al. (2017)) concentrate on facet normal filtering, as facet normal variations offer a more discriminative representation than vertex position variations in describing local geometry. Another important category of mesh denoising techniques is optimization-based methods (Chen et al. (2023)). These methods formulate denoising as an optimization problem and employ strategies such as L0 minimization (He and Schaefer (2013)), sparse regularization (Zhao et al. (2018)), and low-rank matrix recovery (Wei et al. (2018)) to obtain a denoised mesh based on ground-truth geometry or noise pattern priors. Despite their effectiveness, these hand-crafted methods typically require carefully selected parameters and often face challenges when generalized to meshes with diverse geometric features and noise patterns.

To address these issues, data-driven mesh denoising approaches propose to learn parameters or noiseless geometric priors directly from datasets. For example, an early data-driven mesh denoising method (Diebel et al. (2006)) uses a mesh reconstruction model and a feature-enhancing prior learned from noisy data to remove mesh noise. By introducing a fully supervised learning strategy, a cascaded normal regression method (Wang et al. (2016)) formulates mesh denoising as a regression problem to predict noiseless facet normals using filtered facet normal descriptors of a given noisy mesh. Moreover, a cascaded scheme is introduced to improve feature recovery performance further. For mesh denoising regression, graph convolutional networks (GCNs) based mesh denoising approaches (Armando et al. (2020); Shen et al. (2022)) propose a graph structure in the dual space of mesh facets. Armando et al. (2020) use a graph representation of a whole 3D mesh and its multi-scale representation to predict noiseless meshes. On the other hand, Shen et al. (2022) use a graph of aligned local patches for each facet, which can benefit from a data-balancing strategy and aligned facet normals. Surface sampling methods, in addition to the graph representation, are feasible for transforming irregular 3D meshes into regular ones. The local surface descriptors (LSD) method (Zhao et al. (2022)) samples the local region of each facet to create a feature map and then uses the classical ResNet (He et al. (2016)) to learn noiseless facet normals. Existing methods, however, face several challenges in achieving optimal results. The loss of area and distance information in graph-based approaches limits the learning capabilities of regression models. ResNet-based techniques using vanilla CNNs on surfaces introduce rotation ambiguity, limiting their ability to produce more refined outputs.

2.2. Point Cloud Denoising

Point clouds are another important modality for 3D shape representation, widely utilized in fields such as autonomous driving, cultural heritage preservation, and virtual reality. As an essential preprocessing step, point cloud denoising has also garnered substantial research interest. Similar to mesh denoising, point cloud denoising techniques are categorized into three primary types (Zhou et al. (2022)): filter-based, optimization-based, and deep learning-based methods. Filter-based approaches presume noise to be high-frequency, designing specific filters to operate on point positions or normals for denoising. Prominent methods include those based on bilateral filtering (Zhang et al. (2019)), guided filtering (Zheng et al. (2018)), and graph-based techniques (Irfan and Magli (2021)). Optimization-based strategies, on the other hand, treat denoising as an optimization problem, utilizing techniques such as moving least squares (Öztireli et al. (2009)) and sparsity (Mattei and Castrodad (2017)) to reduce noise in point clouds. Both filter and optimization-based methods typically require extensive parameter tuning to achieve optimal results. Conversely, deep learning-based methods directly learn configurations from datasets, employing PointNet (Qi et al. (2017)), GCN (Wei et al. (2023)), and CNN (Wei et al. (2021)) models for point cloud denoising. To adapt point clouds for processing by GCNs and CNNs, common practices include constructing connectivity graphs among adjacent points (Pistilli et al. (2020)) or utilizing projection strategies to project point clouds into 2D images (Lu et al. (2020)). After feature extraction, these models are trained to predict the positions of noise-free points or update point positions to reduce noise (Zhou et al. (2022)). It is noteworthy that, given the shared strategies between point cloud and mesh denoising, our approach offers potential and adaptability in point cloud denoising.

2.3. Deep Learning Image Denoising Strategies

Image denoising aims to reconstruct a clean image from its noisy observation. Over the past decade, deep CNNs have been introduced to solve the image denoising problem as a result of the success of deep learning. According to the training strategy, these image denoising methods can be divided into three categories: 1) real-world noisy images (RNI), 2) additive white-noise images (AWNI), and 3) generated noisy images (GNI) denoising methods. RNI denoising approaches (Zhang et al. (2021)) require real-world noisy-clean image pairs data for training denoising models.

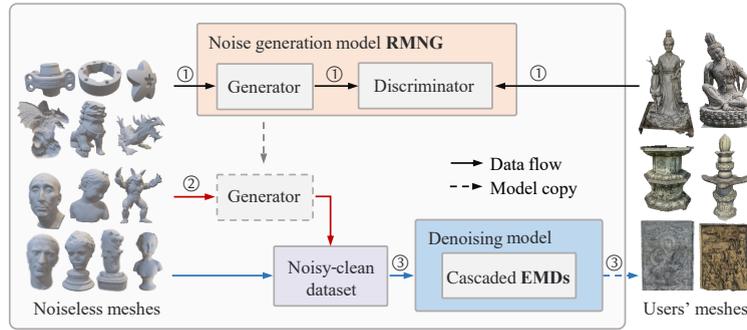


Figure 1: Overview of our method for mesh denoising. For a real-world mesh denoising task such as photogrammetry-derived statue model denoising, we first use the RMNG model to learn noise patterns from user-provided noisy meshes. Then, the trained generator within the RMNG model is used to degrade noiseless meshes to create extensive noisy-clean paired data. Lastly, the generated noisy-clean data is used to train the cascaded EMD models, offering superior denoising performance for the given noisy meshes.

Such datasets can be obtained by recording short-exposure and long-exposure photographs of the same scene, which is expensive and labor-intensive work. Awni denoising methods (Zhang et al. (2017)) use white noise, including Gaussian, Poisson, and salt-and-pepper noise, to degrade high-quality images for paired training data collection. However, since real noise is more complex, models trained on white-noise data frequently perform poorly in real-world denoising tasks. To alleviate this problem, GNI denoising methods (Guo et al. (2019)), which are most related to our work, suggest learning noise features from real-world noisy images. In the image denoising model training phases, noise features are used explicitly (to generate realistic noisy images for training (Cai et al. (2021))) or implicitly (integrating the capability of learning noise features (Wu et al. (2020))).

Extensive experiments (Krull et al. (2019)) have shown that GNI denoising methods learning realistic noise patterns outperform other methods in real-world image denoising tasks. This inspires us to look into creating realistic noise training data to improve the performance of mesh denoising models in real-world tasks. However, due to the irregular connectivity of 3D meshes as opposed to the regular grid structure of images, generating realistic noise patterns for meshes requires additional effort and consideration.

3. Overview

Our realistic noise generation and mesh denoising methods are designed to provide tailored denoising models for various real-world mesh denoising tasks. As shown in Fig. 1, users are only required to provide a handful of noisy meshes from their specific mesh denoising task, such as noisy statue models captured via photogrammetry. Our method proposes to learn the noise patterns inherent in noisy meshes and generates a significant amount of paired noisy-clean data for training the denoising model. First, users' real noisy meshes are represented as noise height maps, and the RMNG model is trained to generate realistic noise maps for existing noiseless meshes. Following this, the generator component of the trained RMNG model is deployed to degrade existing noiseless meshes, thereby producing a substantial volume of paired noisy-clean data. Finally, this generated paired data is utilized to train the cascaded EMD models for mesh denoising, which progressively remove noise from the meshes. Overall, our method offers substantial advantages in user-friendliness and adaptability. With a requirement of minimal input data, our approach learns noise patterns from user-provided meshes and trains a customized denoising model to tackle specific real-world mesh denoising challenges.

4. Equivariant Model for Mesh Denoising (EMD)

Given a noisy mesh and its noiseless counterpart, respectively represented as their corresponding facet normals M^* and M , data-driven mesh denoising approaches can be expressed as: $\hat{M} = f(\phi(M^*))$, where \hat{M} is the predicted noiseless facet normals, $\phi(\cdot)$ is a function depicting the input noisy mesh, and $f(\cdot)$ is a regression model.

Fig. 2 shows the pipeline of the cascaded EMD models for mesh denoising. We first construct $\phi(M^*)$ of the input noisy mesh as facet normals of local regions. Geodesic patches are used to gather the normal signals of facets and

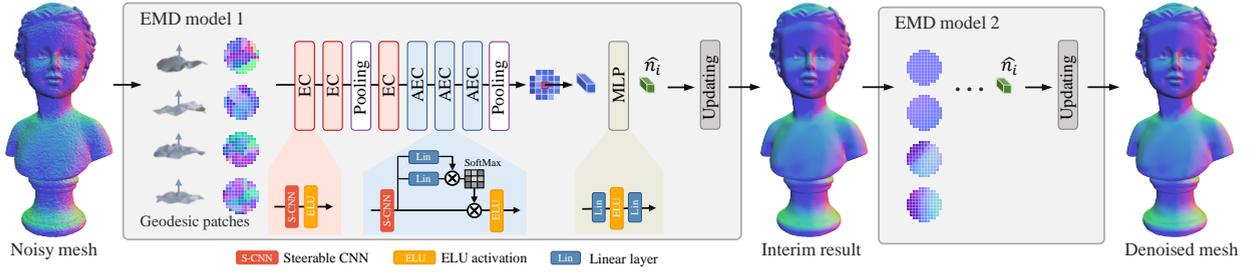


Figure 2: The pipeline of our cascaded EMD models for mesh denoising. Starting with a noisy mesh, we first extract geodesic patches for each facet and align them according to their average normal. These patches, represented as facet normals, are then input into a network made up of EC, AEC, and pooling layers to derive rotation-equivariant feature maps. The central elements of the obtained feature maps are used for noiseless normal prediction. The resulting normals are used to remove noise through vertex updating. We employ a cascaded architecture based on model 2 with the same structure for further noise reduction.

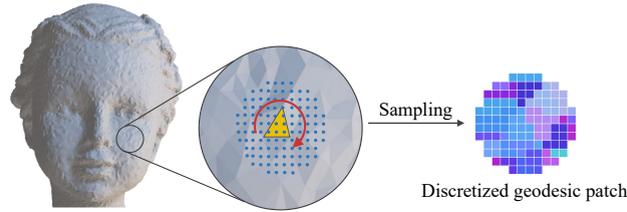


Figure 3: Geodesic patch. Samples are collected on the mesh surface using geodesic paths. Then, the sampled face normals are utilized as pixel values in the geodesic patch representation, with red arrows indicating freely rotatable directions.

their neighbors. Since the geodesic patches have a rotational degree of freedom that can rotate freely on the surface, we introduce an efficient rotation-equivariant model as $f(\cdot)$ to handle the rotated feature map of patches. Then, the rotation-equivariant model is trained to learn the mapping between $\phi(M^*)$ and M , *i.e.*, minimizing $|M - \hat{M}|$. Lastly, we use predicted facet normals for vertex updating and the cascaded regression scheme to progressively remove noise, as described in Wang et al. (2016); Shen et al. (2022).

Geodesic Patch To avoid the loss of facet information caused by constructing the mesh as a graph, we propose using geodesic patches on the surface for representation $\phi(M^*)$. The geodesic patch method is a widely used and effective approach for a local geometric representation in geometry processing works (Masci et al. (2015); Zhao et al. (2022)). For a given 3D mesh, we model a geodesic patch as a disk centered at facet f_i with a radius $r = s \cdot \sqrt{A_i/\pi}$, where A_i represents the total area of the triangles within the 2nd-ring neighborhood of facet f_i , and the scale parameter s , which is set to 2.0 (see Sec. 6.5 for more discussion), controls the radius of geodesic patches.

Discretization and Convolution Our approach involves performing learned convolution kernels k on the surface for noiseless facet normal prediction. In the continuous features on the Euclidean space, the convolution $*$ of patch P_i (the geodesic patch of facet f_i) and the learned kernel k is represented as:

$$(k * P_i)(f_i) = \int_0^{2\pi} \int_0^\rho P_i(r, \theta) k(r, \beta + \theta) dr d\theta, \quad (1)$$

where $\beta \in [0, 2\pi)$ is a phase offset term, and r and θ are polar coordinates. In practice, to perform this convolution operation on the framework of existing deep learning libraries, we need to resample the geodesic patch and convolution kernel and convert the integral into the summation operation. As shown in Fig. 3, we resample the surface using the geodesic path method (Kimmel and Sethian (1998)) to obtain pixel matrices (with resolutions of 28×28) of the geodesic patches and use the faster 2D convolution framework for feature aggregation, similar to the LSD method (Zhao et al. (2022)). To address the spatial transformation learning problem for neural networks, all geodesic patches are rotated and aligned to a common direction by their mean normals (see the left of Fig. 2).

Rotation-Equivariant Model Through resampling the geodesic patch, we obtain the aggregated features for each facet on the noisy mesh. Then, a regression model $f(\cdot)$ is required to learn the mapping between $\phi(M^*)$ and M . Note that there is a phase offset term β in Eq. 1, which represents the phase differences between a geodesic patch and a convolution kernel. The phase offset β is an arbitrary angle in $[0, 2\pi)$ since there is a free rotational degree with the geodesic patch. Employing vanilla CNNs for feature extraction of the rotated patches often suffers from the rotation ambiguity problem and results in blurry outputs (Wiersma et al. (2020)).

To address this issue, recent works in group theory (Worrall et al. (2017)) have investigated equivariant steerable CNNs under symmetry group actions. Moreover, harmonic surface networks (HSN) (Wiersma et al. (2020)) has successfully applied the rotation-equivariant harmonic nets (Worrall et al. (2017)) on surfaces. However, HSN is not applicable to our task since it is performed on discrete vertices instead of facets and requires substantial memory for high-resolution meshes. Therefore, we present an efficient equivariant model to learn rotation-equivariant features for noiseless facet normal prediction. As shown in Fig. 2, our model consists of multiple convolutional, pooling, and linear layers. The convolutional layers include equivariant convolutional (EC) and attention equivariant convolutional (AEC) layers. EC layers adopt a steerable CNN with a general solution of equivariant kernel space constraint proposed by Weiler and Cesa (2019), enabling the extraction of rotation-equivariant features. Meanwhile, AEC layers incorporate a self-attention reweighting module to enhance edge-preserving capacity. Since the attention weight maps are obtained by sample-wise computations, the self-attention module will not disrupt rotational equivariance. Finally, the center element of the feature map, which represents the feature of facet f_i , is extracted and fed into a multilayer perceptron (MLP) for noiseless normal prediction. Additional model details and equivariant model discussions are in the supplementary material.

Normal Refinement and Vertex Updating Minor perturbations and discontinuities are common in predicted facet normals. We use bilateral normal filters to effectively remove these errors, thereby refining the normals for subsequent vertex position updates (Zheng et al. (2010); Shen et al. (2022)). Finally, noise is removed from meshes using the vertex updating method based on a least-squares best-fit solution (Sun et al. (2007)). Additionally, we apply a cascaded strategy utilizing another EMD model (EMD model 2 in Fig. 2) for further noise reduction.

5. Realistic Mesh Noise Generation (RMNG)

Recent image denoising research (Cai et al. (2021); Lee and Kim (2022)) has shown that using additive white noise to degrade noiseless data for the training of data-driven models can result in poor denoising performance in real-world tasks due to the complexity of the intensity and distribution of real noise. On the other hand, collecting large-scale real noisy-clean datasets is expensive and even impractical at times, such as the approach used by Shen et al. (2022), which involves collecting ground-truth meshes, then printing physical 3D models and scanning them for digital meshes. In contrast, we aim to learn realistic noise patterns from given noisy meshes and then degrade existing noiseless meshes using the learned noise patterns to generate massive noisy-clean data pairs. This approach avoids the expensive paired data collection and significantly improves the results of denoising models in real-world denoising tasks.

There are studies (Wu et al. (2020); Cai et al. (2021)) that have explored generating noise on images for creating noisy-clean data. However, unlike images that are represented by regular grids, 3D meshes are curved surfaces with irregular connectivity. As a result, creating realistic noise on meshes necessitates more effort, which should take into account the following factors: 1) The noise patterns should be related to the input meshes; 2) The input and output meshes should maintain a consistent underlying geometry; 3) Captured noisy meshes typically exhibit noise and smooth degradations; 4) Concerning the rotation issue, the discriminator ought to produce rotation-invariant judgments; and 5) The intensity and diversity of the noise patterns should be considered. Taking these factors into account, we propose a generator-discriminator architecture based on steerable per-vertex CNNs, called the realistic mesh noise generation (RMNG) model, to generate realistic noisy-clean datasets, as shown in Fig. 4.

Real Noise Patches In our noise generation RMNG model, two challenges arise when utilizing given noisy meshes as targets: 1) the difficulty in learning noise features due to their lower magnitude compared to geometric information, and 2) inconsistencies between the geometries of input and output meshes from a generator. To address these issues, we eliminate the underlying shapes of noisy meshes, directing RMNG to concentrate on learning noise signals. Given the lack of ground truth for the input noisy meshes, we utilize an EMD model trained on white-noise data to extract the underlying geometry. A potential drawback of this expedient approach might leave some residual geometric infor-

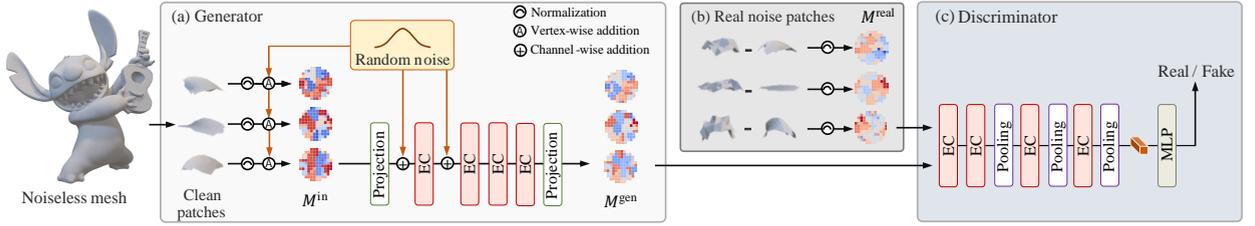


Figure 4: The pipeline of our mesh noise generation model RMNG. We treat mesh noise as vertex height maps, i.e., the distances deviating from the noiseless surface. (a) For the generator, we employ the nearest neighbor algorithm to achieve the geodesic patches of the noiseless mesh as vertex indices and introduce random noise to produce the initial vertex height maps. A network with EC layers then processes the height maps to create realistic noise height maps. (b) The real noise height maps are represented by the differences between the real noisy meshes and their underlying geometry. (c) The discriminator evaluates if the generated noise height maps are as realistic as the real ones.

mation in some real noise patches. Thus, we introduce a zero-mean loss for training, reducing the generation of noise patterns with residual geometry. Additional discussions and visualizations are provided in the supplementary material.

Generator The generator takes the resampled patches of noiseless meshes as input and outputs corresponding realistic noise maps, which are then used to shift noiseless meshes to a noisy state. As shown in Fig. 4 (left), given a noiseless mesh, we first extract the geodesic patches of facets and normalize them to the same size. This procedure is similar to that shown in Fig. 3, except that the samples capture the indices of vertices rather than facets, using the nearest neighbor algorithm based on the Euclidean distance. Then, the vertex index maps are added vertex-wise noise and translated to the vertex position height maps M^{in} that represents the distance of vertex offset from the noiseless mesh. Subsequently, we train a neural network to generate realistic noise maps M^{gen} for the input M^{in} . Random noise is introduced into the channels of feature maps in the early layers to enhance the diversity of the generated noise patterns and to produce variable results. To avoid the rotation ambiguity problem, we also employ the EC layers in the generator. Finally, the generator produces the height maps M^{gen} of the noise patterns, from which we can extract vertex position information for the noisy data.

Discriminator We employ a discriminator to differentiate between samples from the real noise height maps M^{real} and the generated noise height maps M^{gen} , as depicted in Fig. 4 (right). We integrate the EC layers into the discriminator to extract rotation-equivariant features, given that input maps M^{real} and M^{gen} possess rotational freedom. Subsequently, the extracted rotation-equivariant features produce rotation-invariant judgment through the final pooling layer and MLP module. More detailed structures about the generator and discriminator can be found in the supplementary material.

Loss Functions The loss function l_D for the discriminator D only involves differentiating between the height maps of real and generated noise patterns. We use a softplus function for the calculation:

$$l_D(M_i) = \log(1 + \exp(\alpha \cdot D(M_i))), \quad (2)$$

where $\alpha = 1$ for generated patterns, and $\alpha = -1$ for real patterns.

For the loss function l_G of our generator G , we combine the discriminator loss l_{GAN} , zero-mean noise loss l_{zero} , and diversity loss l_{DPP} functions. **First**, the discriminator loss l_{GAN} is an essential loss function for training the GANs generator, expressed as:

$$l_{GAN}(G(P_i, n)) = \mathbb{E}_{n \sim Z} [\log(1 + \exp(-D(G(P_i, n))))], \quad (3)$$

where $G(P_i, n)$ is the output of the generator G with input patch P_i , and n is random noise sampled from a standard normal distribution Z . **Second**, we introduce a zero-mean noise loss l_{zero} defined as:

$$l_{zero}(G(P_i, n)) = \left\| \text{mean}(G(P_i, n)) \right\|. \quad (4)$$

Third, to ensure the diversity of generated noise, we employ the diversity loss l_{DPP} based on the determinantal point processes (DPP) method (Elfeki et al. (2019)). Through calculating the DPP kernel $L = \chi(B)^\top \chi(B)$, where $\chi(\cdot)$

represents the generator without the final layer, and $B = \{M_1, M_2, \dots\}$ is a batch of height maps, the diversity loss l_{DPP} can be expressed as:

$$l_{DPP}(B) = \sum_{t=0}^T \left\| \lambda_t^{\text{real}} - \lambda_t^{\text{gen}} \right\| - \sum_{t=0}^T \hat{\lambda}_t^{\text{real}} \cos(v_t^{\text{real}}, v_t^{\text{gen}}), \quad (5)$$

where λ_t^{real} and λ_t^{gen} are the t^{th} eigenvalues of DPP kernel L for the real and generated noise height maps, respectively, v_t^{real} and v_t^{gen} are the t^{th} eigenvectors of L . **Finally**, the above loss functions are combined for generator loss l_G :

$$l_G = l_{GAN} + w_1 \cdot l_{zero} + w_2 \cdot l_{DPP}, \quad (6)$$

where the w_1 and w_2 are empirically set to 0.5 and 0.5, respectively.

Smooth Degradation 3D meshes captured from the real world frequently exhibit smooth degradation issues. In a manner akin to geometry-recovery approaches (Wang et al. (2019)), we address this challenge by applying cotangent Laplacian smoothing to the input meshes during the generation of noisy-clean training data:

$$v'_i = v_i + \sum_{\{i,j\} \in \mathbf{E}} w_{ij}(v_j - v_i), \quad (7)$$

where v_i and v'_i are the original and updated vertices, \mathbf{E} is the set of edges connected to v_i , and w_{ij} are the cotangent weights of edge $\{i, j\}$ (Nealen et al. (2006)). The input meshes undergo 0, 1, and 2 iterations of Laplacian smoothing to achieve various levels of smooth degradation. Training on smoothed meshes helps the denoising model to gain specific feature recovery capabilities, resulting in aesthetically appealing denoising outputs.

Noisy-clean Dataset Generation To extract paired noisy-clean datasets from the trained RMNG model, we first represent the smoothed noiseless meshes as geodesic patches of vertices. These patches are fed into the RMNG model to generate corresponding noise patterns in the vertex height maps. Next, we extract vertex displacements from the height maps and apply them to vertices along normals, producing noisy patches. This procedure creates realistic noisy patches for training denoising models. Besides, We augment the generated noise signals by scaling the vertex displacements with factors of 1 and 2 during denoising training data generation to enhance the robustness of trained denoising models.

6. Experiments

6.1. Datasets and Metrics

To evaluate the mesh denoising results, we use the synthetic datasets (SysData), real scanned datasets (ScanData) (Wang et al. (2016)), and print datasets (PrintData) (Shen et al. (2022)). The SysData comprises 208 models in three types: CAD models, smooth models, and models with rich features. The training and test data are obtained by adding white noise at three different levels to noiseless meshes. The ScanData consists of 1) Kv1Data, which contains 73 models scanned by Microsoft Kinect v1; and 2) Kv2Data, which includes 72 models scanned by Microsoft Kinect v2.

In addition, we evaluate the methods on a new mesh dataset (see Fig. 6) captured by high-end cameras and photogrammetry techniques, called the photogrammetry dataset (PhgmData). It contains eight high-resolution statue models with various noise patterns. The noise may be introduced by the uneven surfaces of the statues, image noise from the captured photos, and noise from the reconstruction process. These models are intended for use in cultural heritage and video game asset production, offering an in-depth evaluation of mesh denoising approaches in real-world tasks.

Implementation Details Our experiments are conducted on a desktop PC with an NVIDIA GeForce RTX 3090 GPU, an Intel i7-11700K CPU (3.60GHz), and 64GB memory. For optimization, we use the Adam optimizer with a learning rate of 10^{-3} , betas of (0.9, 0.999), and eps set to 10^{-4} . The batch size for training is 500, and we train the RMNG model for 40 epochs and the EMD model for 50 epochs.

6.2. Denoising Models Comparisons (EMD model Only)

We first separately assess the denoising performance of our EMD model on existing datasets with ground truth, and compare it with the SOTA methods, including bilateral normal filtering (BNF) (Zheng et al. (2010)), L0 smoothing

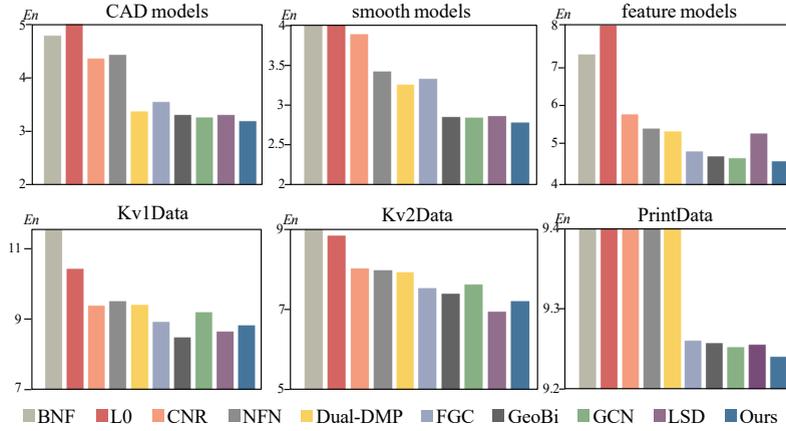


Figure 5: Comparisons with SOTA methods on the SysData (including CAD, smooth, and feature models), Kv1Data, Kv2Data, and PrintData datasets using the average angular difference E_n metric. Higher bars are truncated for better illustration.

(L0) (He and Schaefer (2013)), cascaded normal regression (CNR) (Wang et al. (2016)), NormalF-Net (NFN) (Li et al. (2020)), dual deep mesh prior (Dual-DMP) (Hattori et al. (2022)), facet graph convolutions (FGC) (Armando et al. (2020)), GeoBi-GNN (GeoBi) (Zhang et al. (2022)), GCN-Denoiser (GCN) (Shen et al. (2022)), and local surface descriptor (LSD) (Zhao et al. (2022)). We use the denoised results or trained models of CNR, NFN, FGC, GeoBi, GCN, and LSD methods provided by the original authors. For the outcomes of BNF and L0, we employed fine-tuned parameters to generate the best possible results. Note that the PrintData dataset lacks noisy-clean paired data, making direct training of data-driven models on this dataset infeasible. Therefore, we follow the approach employed by the authors of PrintData (Shen et al. (2022)), using models previously trained on the SysData dataset to perform denoising inference on PrintData for each method.

The average angular difference E_n (Wang et al. (2016)) between the facet normals of a denoised mesh and its ground-truth mesh is employed to evaluate our denoising results and compare our method with the SOTA methods. Fig. 5 presents the quantitative comparisons between our method and other approaches on the SysData, ScanData, and PrintData datasets. It is clear that data-driven methods surpass traditional filtering and optimization-based approaches; the learning self-prior method, Dual-DMP, achieves results that are generally inferior to the SOTA fully supervised methods. The results demonstrate that our method outperforms competing approaches across these datasets, especially for feature-rich models in “*feature models*” and “*PrintData*”. Our results on Kv1Data and Kv2Data are slightly inferior to LSD. This discrepancy can be attributed to that LSD method employs a sampling point count 6 times greater than ours and uses a larger network model, leading to significantly slower execution. Furthermore, these two datasets only contain smooth geometries, limiting their capacity to exhibit the feature-preserving capability of the denoising algorithms.

In Tab. 1, we report the runtime of various learning-based denoising methods on test models. CNR achieves the shortest runtime but underperforms compared to other methods (see Fig. 5). Our model’s runtime is slower than that of GCN and Dual-DMP; however, this marginal difference is generally acceptable. Conversely, our model is approximately six times faster than the LSD method, which relies on very dense sampling points and a complex prediction model. Overall, our model offers a well-balanced trade-off between efficacy and efficiency.

6.3. Generated Noise Effectiveness (Our Full Method)

Subsequently, we assess the effectiveness of the RMNG generated training data on the PrintData and PhgmData. PrintData contains real noise from 3D printing and scanning processes, while PhgmData includes noise from imaging and photogrammetry methods. These noise signals represent the types of noise that would be encountered in real-world scenarios, and thus, these two datasets are chosen for our evaluation. We conduct our experiments independently on each of the two datasets, separately training both the noise generator and the denoising model.

The PrintData dataset is randomly divided into training and test sets, each containing 10 meshes. For the PhgmData dataset, 4 captured noisy models (first row of Fig. 6) are used to train RMNG, and the remaining 4 models (second

Table 1

Runtime comparison of CNR, GCN, Dual-DMP, LSD, and our denoising method on some test meshes, including both data preprocessing and model inference times. Time is measured in seconds. Note that the runtime for the GCN method is measured on an i7-8700K CPU and NVIDIA GTX 1080Ti GPU because the source code relies on CUDA 10, which does not run successfully on newer GPUs; other methods are measured on the device mentioned in Sec. 6.1.

Model	Number of faces	CNR	GCN	Dual-DMP	LSD	Ours
Gargoyle	171,112	9.39s	139.20s	134.15s	1632.60s	245.81s
Turbine Lp	114,040	5.57s	70.14s	78.33s	940.12s	146.27s
Nicolo	99,994	4.80s	63.04s	72.36s	787.22s	128.70s
Joint	41,808	2.31s	29.24s	41.24s	338.40s	64.28s

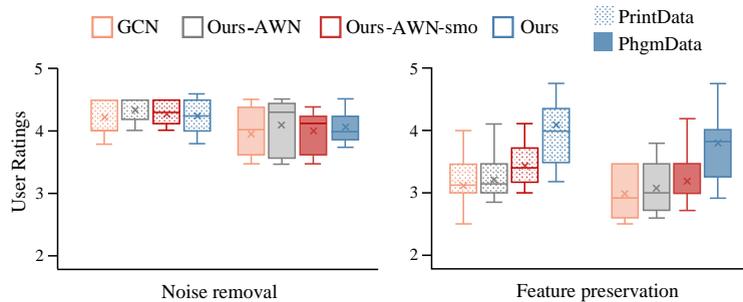
**Figure 6:** The PhgmData dataset contains feature-rich meshes.

Figure 7: User study results. Professional artists rate the denoising results based on 1) noise removal and 2) geometric feature preservation performance. The rating scale ranges from 1 to 5, with higher scores indicating better performance. “Ours” method demonstrates superior geometric feature preservation performance.

row of Fig. 6) are reserved for testing. We also employ the noiseless models from SysData to generate paired training data. On these two datasets, we compare our full method (“Ours”) with our EMD model trained on smoothed data with additive white noise (“Ours-AWN-smo”), our EMD model trained on the additive white noise data (“Ours-AWN”), and the other methods FGC (Armando et al. (2020)) and GCN (Shen et al. (2022)) trained on the additive white noise data (SysData). The comparison results are presented in Fig. 8. It is evident that when trained on the AWN datasets, the feature preservation performance of FGC, GCN, and “Ours-AWN” is relatively inferior. Although “Ours-AWN-smo”

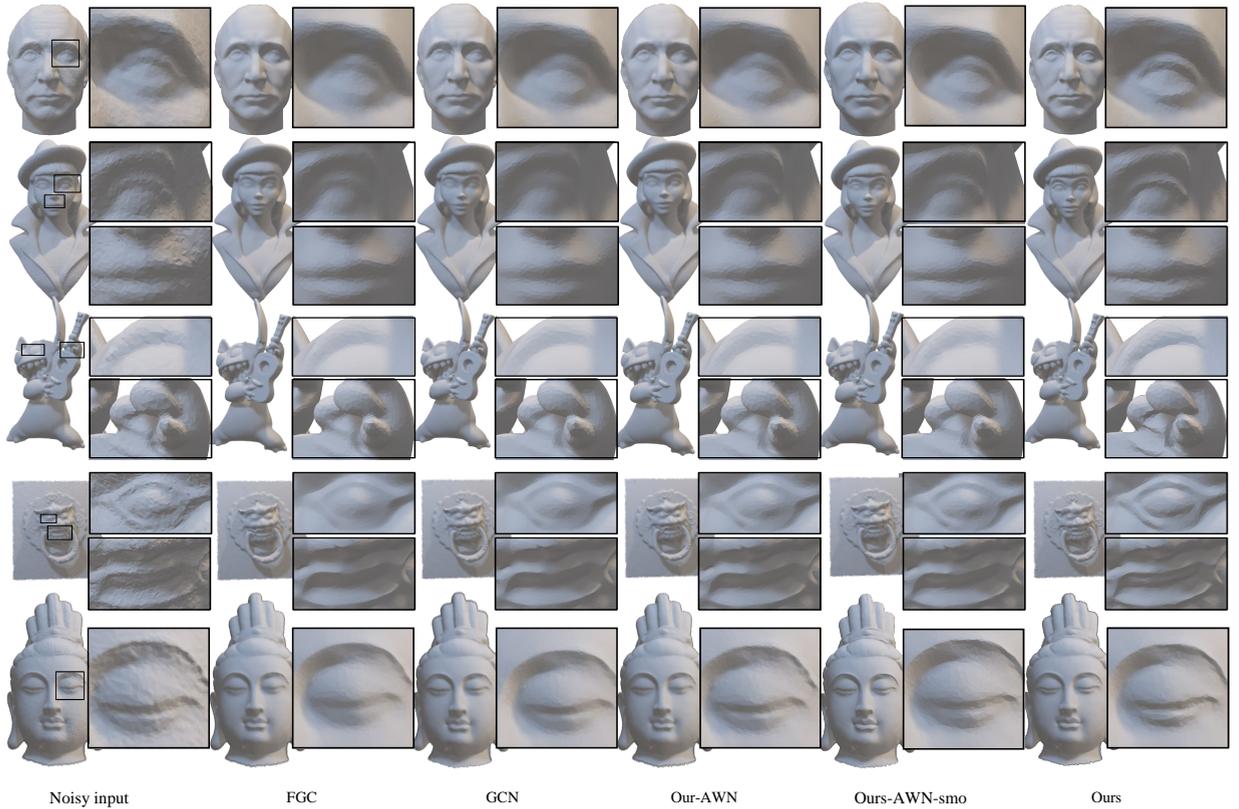


Figure 8: Comparisons of denoising methods on real-world captured meshes in PrintData and PhgmData. From left to right: noisy input, denoised results of FGC (Armando et al. (2020)), GCN (Shen et al. (2022)), our model trained on additive white noise (AWN) data, trained on AWN and smooth data, and trained on data generated by the RMNG.

shows some improvement, it still falls short compared to our full method. “Ours” exhibits significant enhancements in feature preservation and recovery. Many subtle details, which are lost in the models trained on the AWN dataset, are effectively preserved by the “Ours” method.

Furthermore, we conduct a user study to evaluate the denoising results from two perspectives: 1) noise removal performance and 2) geometric feature preservation and recovery. We invite ten professional video game artists with three to ten years of experience in 3D modeling and mesh repair to participate in the user study. As a result, they are uniquely qualified to evaluate the performance of denoising models. They are asked to rate the denoising results on a scale of 0 to 5, with higher scores indicating better performance. The results, displayed in Fig. 7, show that all four methods perform well in noise removal, reflected in average scores of approximately 4.3 for PrintData and 4.1 for PhgmData. However, for preserving the geometric features aspect, “Ours” receives significantly higher ratings, with average scores of 4.1 for PrintData and 3.8 for PhgmData. These results outperform those by the AWN approaches, GCN and “Ours-AWN”, with 3.2 and 3.3 for PrintData, and 3.0 and 3.1 for PhgmData, respectively. Although integrating smooth degradation into the white noise training data (“Ours-AWN-smo”) slightly improves the feature recovery performance of the denoising model, there is still a considerable performance gap compared to our full method, with score differences of 0.6 for PrintData and 0.5 for PhgmData. Our full method using the smoothing operation and the RMNG model to generate training data produces superior denoising results, particularly in preserving and recovering geometric details, which is especially useful for professionals in the 3D modeling industry.

6.4. Comparisons Against Learning Self-Prior Method

In addition to AWN-based methods and our proposed generating realistic noise, Dual-DMP (Hattori et al. (2022)), which utilizes deep learning models to learn a self-prior for mesh denoising, also eliminates the need for collecting expensive paired noisy-clean datasets. This method learns the denoising prior from a single input mesh, enabling

Table 2

Quantitative comparison of denoising strategies: learning self-prior (Dual-DMP, Hattori et al. (2022)), AWN-based (GCN, Shen et al. (2022)), and our proposed method on the PrintData dataset, using the metric of average angular difference E_n . The best results are highlighted in **bold**.

Model	Dual-DMP	GCN	Ours	Model	Dual-DMP	GCN	Ours
Minion ghost	5.56	4.64	4.58	Spaghetti detective	8.97	8.21	7.91
Nut	15.35	15.01	15.50	Stitch guitar	6.30	5.84	5.83
Putin	9.55	9.35	8.87	Stitch stand	5.37	4.84	4.79
Snoopy	7.79	6.67	6.55	Suit man	8.40	8.39	8.17
Snoopy flying face	7.79	7.10	7.05	Tp	21.26	20.52	21.69

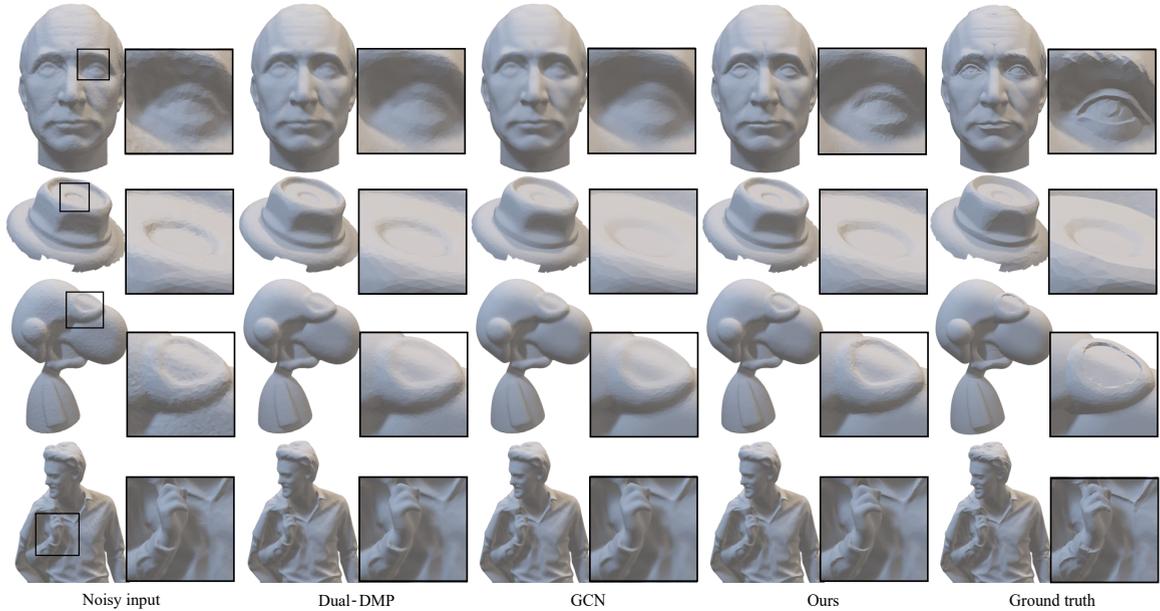


Figure 9: Visual comparison on the PrintData dataset. From left to right: noisy input, self-prior learning (Dual-DMP, Hattori et al. (2022)), AWN-based (GCN, Shen et al. (2022)), our method, and ground truth. Our approach demonstrates enhanced preservation of geometric details.

specialized denoising results for each individual shape. Here, we evaluate the performance of these three denoising strategies: AWN-based (GCN, Shen et al. (2022)), learning self-prior (Dual-DMP, Hattori et al. (2022)), and our proposed method. Quantitative comparisons on the PrintData dataset, as illustrated in Tab. 2, reveal that our method outperforms other methods in denoising 3D meshes obtained from the real world, achieving lower average angular differences across most meshes. The visual comparisons shown in Fig. 9 corroborate these findings. Both the AWN-based and learning self-prior methods often result in the loss of details during the denoising process, whereas our method more effectively preserves the geometric features of the original mesh.

6.5. Ablation Studies

We conduct ablation studies to evaluate the contribution of each component in our method. These experiments provide insights into the performance and validate the effectiveness of our method.

Patch Size for EMD Model The patch size is crucial in the patch-based denoising process, as it defines the local neighborhood used for feature extraction and noise reduction. For geodesic patches, we define a parameter s , which controls the relative size of geodesic patches. We experiment with different patch sizes by varying the value of s and evaluate the denoising performance using the E_n metric on the SysData, Kv1Data, and Kv2Data datasets. As shown in Fig. 10, the best results are achieved when s lies between 2 and 4. Consequently, we set $s = 2.0$ for our method.

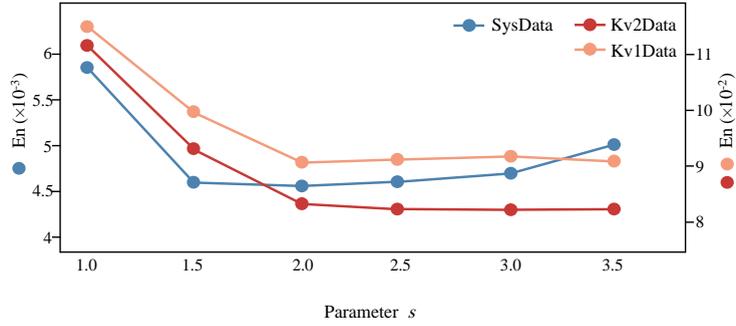


Figure 10: Ablation experiments for the geodesic patch size parameter s on the SysData, Kv1Data, and Kv2Data datasets.

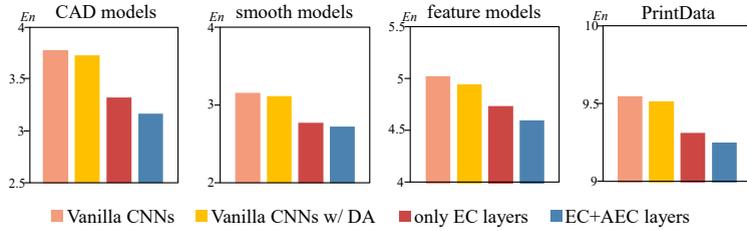


Figure 11: Ablation experiments for equivariant convolution (EC) and equivariant convolution with attention (AEC). We test the denoising models with vanilla CNNs, only using EC layers, and using both EC and AEC layers.

Equivariant Layer and Attention Module To evaluate the effectiveness of the equivariant layers EC and AEC, we conduct an ablation study with four versions of our model: one with vanilla CNNs, one with vanilla CNNs that trained on a dataset with rotated data augmentation, one with only EC layers, and one with EC and AEC layers. These models are trained on the SysData dataset, and the denoising results are assessed using the E_n metric. The results in Fig. 11 show that the models with rotation-equivariant convolution layers consistently outperform the one with vanilla CNNs. Furthermore, the model with attention weights achieves better results, demonstrating our AEC layers' effectiveness.

7. Conclusions

This paper has presented a rotation-Equivariant model based Mesh Denoising (EMD) method and a Realistic Mesh Noise Generation (RMNG) method to address the challenges associated with denoising captured feature-rich meshes. Our EMD method, leveraging rotation-equivariant features and self-attention weights, achieves SOTA results in preserving accurate underlying features while removing noise. The RMNG method, grounded in the GANs architecture, generates massive realistic noisy-clean mesh pairs for training the denoising model. We endow the trained denoising model with feature recovery capabilities by incorporating Laplacian smoothing into training data generation. This strategy enables more effective denoising in real-world scenarios where smoothed features may impact the quality of captured meshes. Experimental results demonstrate that our method notably retains and recovers fine-grained features in real-world mesh denoising tasks.

Acknowledgements

Xiaogang Jin was supported by the Key R&D Program of Zhejiang (No. 2023C01047), the National Natural Science Foundation of China (Grant No. 62036010), and the FDCT under Grant 0002/2023/AKP.

References

Armando, M., Franco, J.S., Boyer, E., 2020. Mesh denoising with facet graph convolutions. *IEEE Transactions on Visualization and Computer Graphics* 28, 2999–3012.

- Cai, Y., Hu, X., Wang, H., Zhang, Y., Pfister, H., Wei, D., 2021. Learning to generate realistic noisy images via pixel-level noise-aware adversarial training. *Advances in Neural Information Processing Systems* 34, 3259–3270.
- Chen, H., Li, Z., Wei, M., Wang, J., 2023. Geometric and learning-based mesh denoising: a comprehensive survey. *ACM Transactions on Multimedia Computing, Communications and Applications* 20, 1–28.
- Clarenz, U., Diewald, U., Rumpf, M., 2000. Anisotropic geometric diffusion in surface processing, in: *Proceedings of the Conference on Visualization '00*, pp. 397–405.
- Desbrun, M., Meyer, M., Schröder, P., Barr, A.H., 1999. Implicit fairing of irregular meshes using diffusion and curvature flow, in: *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 317–324.
- Diebel, J.R., Thrun, S., Brünig, M., 2006. A Bayesian method for probable surface reconstruction and decimation. *ACM Transactions on Graphics (ToG)* 25, 39–59.
- Elfeki, M., Couprie, C., Riviere, M., Elhoseiny, M., 2019. GDPP: Learning diverse generations using determinantal point processes, in: *Proceedings of the 36th International Conference on Machine Learning*, pp. 1774–1783.
- Guo, S., Yan, Z., Zhang, K., Zuo, W., Zhang, L., 2019. Toward convolutional blind denoising of real photographs, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1712–1722.
- Hattori, S., Yatagawa, T., Ohtake, Y., Suzuki, H., 2022. Learning self-prior for mesh denoising using dual graph convolutional networks, in: *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, Springer, pp. 363–379.
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 770–778.
- He, L., Schaefer, S., 2013. Mesh denoising via L0 minimization. *ACM Transactions on Graphics (ToG)* 32, 1–8.
- Irfan, M.A., Magli, E., 2021. Exploiting color for graph-based 3D point cloud denoising. *Journal of Visual Communication and Image Representation* 75, 103027.
- Kimmel, R., Sethian, J.A., 1998. Computing geodesic paths on manifolds. *Proceedings of the National Academy of Sciences* 95, 8431–8435.
- Krull, A., Buchholz, T.O., Jug, F., 2019. Noise2Void-learning denoising from single noisy images, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2129–2137.
- Lee, K.W., Wang, W.P., 2005. Feature-preserving mesh denoising via bilateral normal filtering, in: *Ninth International Conference on Computer Aided Design and Computer Graphics (CAD-CG'05)*, pp. 6–pp.
- Lee, S., Kim, T.H., 2022. NoiseTransfer: Image noise generation with contrastive embeddings, in: *Proceedings of the Asian Conference on Computer Vision*, pp. 3569–3585.
- Li, Z., Zhang, Y., Feng, Y., Xie, X., Wang, Q., Wei, M., Heng, P.A., 2020. Normalf-Net: Normal filtering neural network for feature-preserving mesh denoising. *Computer-Aided Design* 127, 102861.
- Lu, D., Lu, X., Sun, Y., Wang, J., 2020. Deep feature-preserving normal estimation for point cloud filtering. *Computer-Aided Design* 125, 102860.
- Lu, X., Chen, W., Schaefer, S., 2017. Robust mesh denoising via vertex pre-filtering and L1-median normal filtering. *Computer Aided Geometric Design* 54, 49–60.
- Masci, J., Boscaini, D., Bronstein, M., Vandergheynst, P., 2015. Geodesic convolutional neural networks on Riemannian manifolds, in: *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 37–45.
- Mattei, E., Castrodad, A., 2017. Point cloud denoising via moving RPCA, in: *Computer Graphics Forum*, Wiley Online Library, pp. 123–137.
- Nealen, A., Igarashi, T., Sorkine, O., Alexa, M., 2006. Laplacian mesh optimization, in: *Proceedings of the 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia*, pp. 381–389.
- Nocerino, E., Stathopoulou, E.K., Rigon, S., Remondino, F., 2020. Surface reconstruction assessment in photogrammetric applications. *Sensors* 20, 5863.
- Öztireli, A.C., Guennebaud, G., Gross, M., 2009. Feature preserving point set surfaces based on non-linear kernel regression, in: *Computer Graphics Forum*, Wiley Online Library, pp. 493–501.
- Pistilli, F., Fracastoro, G., Valsesia, D., Magli, E., 2020. Learning graph-convolutional representations for point cloud denoising, in: *European Conference on Computer Vision*, Springer, pp. 103–118.
- Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017. Pointnet: Deep learning on point sets for 3D classification and segmentation, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 652–660.
- Shen, Y., Fu, H., Du, Z., Chen, X., Burnaev, E., Zorin, D., Zhou, K., Zheng, Y., 2022. GCN-Denoiser: Mesh denoising with graph convolutional networks. *ACM Transactions on Graphics (ToG)* 41, 1–14.
- Sun, X., Rosin, P.L., Martin, R., Langbein, F., 2007. Fast and effective feature-preserving mesh denoising. *IEEE Transactions on Visualization and Computer Graphics* 13, 925–938.
- Wang, J., Huang, J., Wang, F.L., Wei, M., Xie, H., Qin, J., 2019. Data-driven geometry-recovering mesh denoising. *Computer-Aided Design* 114, 133–142.
- Wang, P.S., Liu, Y., Tong, X., 2016. Mesh denoising via cascaded normal regression. *ACM Transactions on Graphics (ToG)* 35, 1–12.
- Wei, M., Chen, H., Zhang, Y., Xie, H., Guo, Y., Wang, J., 2021. GeoDualCNN: Geometry-supporting dual convolutional neural network for noisy point clouds. *IEEE Transactions on Visualization and Computer Graphics* 29, 1357–1370.
- Wei, M., Huang, J., Xie, X., Liu, L., Wang, J., Qin, J., 2018. Mesh denoising guided by patch normal co-filtering via kernel low-rank recovery. *IEEE Transactions on Visualization and Computer Graphics* 25, 2910–2926.
- Wei, M., Wei, Z., Zhou, H., Hu, F., Si, H., Chen, Z., Zhu, Z., Qiu, J., Yan, X., Guo, Y., et al., 2023. AGConv: Adaptive graph convolution on 3D point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 9374–9392.
- Wei, M., Yu, J., Pang, W.M., Wang, J., Qin, J., Liu, L., Heng, P.A., 2014. Bi-normal filtering for mesh denoising. *IEEE Transactions on Visualization and Computer Graphics* 21, 43–55.
- Weiler, M., Cesa, G., 2019. General E(2)-Equivariant Steerable CNNs, in: *Conference on Neural Information Processing Systems (NeurIPS)*.
- Wiersma, R., Eisemann, E., Hildebrandt, K., 2020. CNNs on surfaces using rotation-equivariant features. *ACM Transactions on Graphics (ToG)*

39.

- Worrall, D.E., Garbin, S.J., Turmukhambetov, D., Brostow, G.J., 2017. Harmonic networks: Deep translation and rotation equivariance, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5028–5037.
- Wu, X., Liu, M., Cao, Y., Ren, D., Zuo, W., 2020. Unpaired learning of deep image denoising, in: Computer Vision – ECCV 2020, Springer. pp. 352–368.
- Zhang, F., Zhang, C., Yang, H., Zhao, L., 2019. Point cloud denoising with principal component analysis and a novel bilateral filter. *Traitement du Signal* 36, 393–398.
- Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L., 2017. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing* 26, 3142–3155.
- Zhang, T., Fu, Y., Li, C., 2021. Hyperspectral image denoising with realistic data, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 2248–2257.
- Zhang, Y., Shen, G., Wang, Q., Qian, Y., Wei, M., Qin, J., 2022. GeoBi-GNN: Geometry-aware bi-domain mesh denoising via graph neural networks. *Computer-Aided Design* 144, 103154.
- Zhao, W., Liu, X., Jiang, J., Zhao, D., Li, G., Ji, X., 2022. Local surface descriptor for geometry and feature preserved mesh denoising, pp. 3446–3453.
- Zhao, Y., Qin, H., Zeng, X., Xu, J., Dong, J., 2018. Robust and effective mesh denoising using L0 sparse regularization. *Computer-Aided Design* 101, 82–97.
- Zheng, Y., Fu, H., Au, O.K.C., Tai, C.L., 2010. Bilateral normal filtering for mesh denoising. *IEEE Transactions on Visualization and Computer Graphics* 17, 1521–1530.
- Zheng, Y., Li, G., Xu, X., Wu, S., Nie, Y., 2018. Rolling normal filtering for point clouds. *Computer Aided Geometric Design* 62, 16–28.
- Zhou, L., Sun, G., Li, Y., Li, W., Su, Z., 2022. Point cloud denoising review: from classical to deep learning-based approaches. *Graphical Models* 121, 101140.