Topological and Node Noise Filtering on 3D Meshes Using Graph Neural Networks (Student Abstract)

Vladimir Mashurov[1, 2], **Natalia Semenova**[1, 3]
[1]Sberbank PJSC
[2]ITMO University
[3]AIRI
Moscow, 117997 Russia
mashurovvladimirv@gmail.com, semenova.bnl@gmail.com

Abstract

Topological and node noise filtration are typically considered separately. Graph Neural Networks (GNN) are commonly used for node noise filtration, as they offer high efficiency and low exploitation costs. This paper explores the solution of joint node and topological noise filtration through the use of graph neural networks. Since treating a 3D mesh as a graph is challenging, an indicator function grid representation is employed as input for GNNs to perform the joint filtering. The resulting machine learning model is inspired by point cloud to mesh reconstruction algorithms and demonstrates low computational requirements during inference, producing successful results for smooth, watertight 3D models.

Introduction

The problem of mesh denoising occurs in various fields whenever there is a need to reconstruct a digital copy of a real-world object. Due to imperfections in the instruments and scanners used to capture information about the object's geometry, the resulting 3D models often contain a significant amount of noise, which hinders their accuracy as representations of the original object.

3d-mesh denoising task was studied from various perspectives regarding the type of 3d-models: CAD (Zhang et al. 2015; He and Schaefer 2013) and nonCAD (Zheng et al. 2011) models, or time efficiency (Sun et al. 2007). However, these methods only work with one type of noise that commonly occurs in practice. This type of noise affects the positions of mesh nodes but does not compromise the integrity of the mesh. Another type is called topological noise, which occurs as the absence of nodes and faces. This task is also called a hole filling (Sarkar, Varanasi, and Stricker 2018) and borders with mesh reconstruction tasks.

This work focuses on a solution that deals with the dual problem of node and topological noise removal. The suggested method converts 3d-mesh into a signed distance field as indicator function grid (Peng et al. 2021). Then, it updates the values in grid nodes via GNN to obtain a better version of a mesh after applying marching cubes' algorithm (Lorensen and Cline 1987).

Related Work

The initial attempts to address the issue of node noise filtering involved deterministic algorithms (Zheng et al. 2011; Zhang et al. 2015; He and Schaefer 2013). However, their usability was limited because these methods could not encompass the full range of mesh geometries. Later, data-driven methods (Wang, Liu, and Tong 2016) demonstrated improved filtration quality and versatility. Recent studies show that deep learning models based on GNNs (Shen et al. 2022; Zhang et al. 2022) perform the best in this case.

In the field of topological noise filtration, deterministic approaches also have limitations when it comes to the presence of structure regularity (Pauly et al. 2008). Machine learning approaches can solve the narrow problem of hole filling or mesh reconstruction (Sarkar, Varanasi, and Stricker 2018). These tasks can be theoretically considered as mesh noise filtering in general. Drawing a distinct border between the task of topological noise filtration and mesh reconstruction is challenging.

Method

The proposed method is inspired by (Peng et al. 2021), the method leverages its reconstruction abilities for denoising task and consists of three transformations as: $a(X) = p_{out}(\mu_\theta(p_{in}(X)))$. The first stage p_{in} is conversion of a mesh to indicator function grid of size 128^3. The second step μ_θ comprises three GNNs and one MLP layer to modify the values in nodes of the grid, θ is a learnable parameters' set. The last part consist of p_{out} - marching cubes algorithm to reconstruct the denoised mesh out of the resulting grid. The pipeline is depicted on Figure 1.

Model is trained on NVIDIA V100 GPU, the PyTorch framework is used for implementation, along with the Adam optimizer. Learning rate is set to 0.001, batch size is 1 and dropout probability is 0.1. Inference tests conducted on a computer with 16 GB of RAM, Intel Core(TM) i7-11370H CPU and without GPU. The demo and code are available on Hugging Face[1].

Experiments & Results

In the experiments, topological noise is generated by equally probable removal of a given percent of vertices. The metrics

[1]https://huggingface.co/spaces/MVV/3dTopDenoising

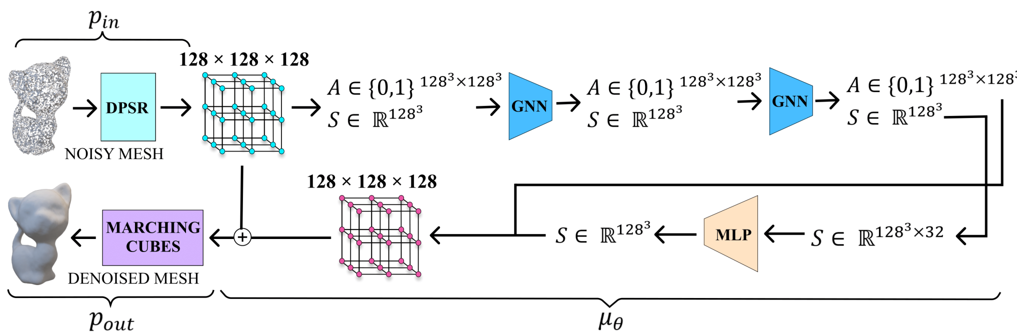

Figure 1: The model pipeline, where A is the grid adjacency matrix, and S is the grid nodes' features matrix.

method/metric	VCD	NMCD	AAD	AVD
2GATv2	11,06	16,6319	7,2706	6,958
GeoBiGNN	0,16	**0,5264**	**0,9239**	1,206
CascadedReg	0,15	1,0154	1,2366	1,268
GuidedMesh	0,17	1,9120	1,2297	**1,141**
BilaterNorm	**0,11**	1,3201	1,2878	1,165
Fast&Effective	0,35	3,6027	2,5981	2,104

Table 1: Comparison of filtering nodal noise models on the Synthetic dataset from paper (Wang, Liu, and Tong 2016). VCD – vertices CD ($\times 10^{-4}$), NMCD – normals' mean cosine distance($\times 10^{-2}$), AAD – absolute area difference($\times 10^{-2}$), AVD – absolute volume difference($\times 10^{-3}$)

were estimated using normalized data, so each mesh was adjusted to fit into unit-cube on the positive octant. To train the solution proposed in this paper, the MSE metric is used as:

$$MSE = \frac{1}{|G_x|} \sum_{v_x \in G_x} (v_x - v_y)^2,$$

where G_x and G_y are resulting and ground truth indicator function grids, v_x and v_y are corresponding values in grids' nodes.

Table 1 presented the comparison of the proposed model with other approaches in the node noise filtering task. For additional verification, the Chamfer distance metric between meshes' nodes is used, it can be written as follows:

$$CD(X,Y) = \sum_{x \in X} \min_{y \in Y} \|x - y\|_2^2 + \sum_{y \in Y} \min_{x \in X} \|x - y\|_2^2,$$

where X and Y are sets of resulting and ground truth meshes' nodes. $\|\|_2$ is L_2 norm. The normals' mean cosine distance can be derived as:

$$NMCD = \frac{1}{|F_x|} \sum_{n_x \in F_x} \left(1 - \frac{n_x n_y}{\|n_x\|_2 \|n_y\|_2}\right),$$

where F_x and F_y are sets of resulting and ground truth meshes faces, n_x and n_y are normal vectors of the corresponding face in F_x and F_y. Despite the fact that model quality in this setting is lower than other methods, given methods are not capable of topological noise filtering. Besides, the quality is sufficient to remove visible nodal noise. Among deterministic filters, BilaterNorm filter (Zheng et al. 2011) have the best result due to its versatility, but GuidedMesh filter (Zhang et al. 2015) shows better results on CAD models. Fast&Effective filter (Sun et al. 2007) sacrifices the quality of filtering to efficiency. CascadedReg (Wang, Liu, and Tong 2016) works better on smooth 3D models without corners and sharp edges.

The ablation study showed that GATv2 (Brody, Alon, and Yahav 2022) graph network layers perform best in the presented model, two GATv2 layers are good enough for reaching the quality established in this research. The smoothing coefficient of DPSR (Peng et al. 2021) is set to 3.0, the higher values negatively affect model's quality. The constraints include the capacity to successfully filter only watertight meshes and the difficulties of filtering CAD models. The latter is caused by the Gibbs effect and the inability to clearly represent sharp geometry in indicator function grid representation.

Discussion

Recent studies have demonstrated that deep diffusion models (Zeng et al. 2022) and voxelization-based approaches have the potential to successfully transform a latent representation of a 3D model. However, both of these methods are computationally expensive. The use of indicator function grid representation shows promise for combining topological and node filtration in a joint manner. This representation provides a graph structure that can be processed by GNNs. For future work, it is essential to test the approach on octrees, which are more suitable for graph density than grids. Additionally, the network should be adapted to modify the distance function and texture.

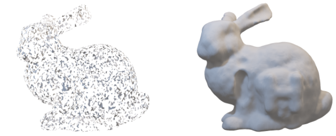

Figure 2: Example of mesh denosing, the data used from (Wang, Liu, and Tong 2016). "Bunny Hi", 50% of vertices are preserved, low node noise.

References

Brody, S.; Alon, U.; and Yahav, E. 2022. How Attentive are Graph Attention Networks? In *International Conference on Learning Representations*.

He, L.; and Schaefer, S. 2013. Mesh Denoising via L0 Minimization. *ACM Trans. Graph.*, 32(4).

Lorensen, W. E.; and Cline, H. E. 1987. Marching cubes: A high resolution 3D surface construction algorithm. *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*.

Pauly, M.; Mitra, N. J.; Wallner, J.; Pottmann, H.; and Guibas, L. J. 2008. Discovering Structural Regularity in 3D Geometry. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08. New York, NY, USA: Association for Computing Machinery. ISBN 9781450301121.

Peng, S.; Jiang, C. M.; Liao, Y.; Niemeyer, M.; Pollefeys, M.; and Geiger, A. 2021. Shape As Points: A Differentiable Poisson Solver. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Sarkar, K.; Varanasi, K.; and Stricker, D. 2018. 3D Shape Processing by Convolutional Denoising Autoencoders on Local Patches. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1925–1934.

Shen, y.; Fu, H.; Du, Z.; Chen, X.; Burnaev, E.; Zorin, D.; Zhou, K.; and Zheng, Y. 2022. GCN-Denoiser: Mesh Denoising with Graph Convolutional Networks. *ACM Trans. Graph.*, 41(1).

Sun, X.; Rosin, P. L.; Martin, R.; and Langbein, F. 2007. Fast and Effective Feature-Preserving Mesh Denoising. *IEEE Transactions on Visualization and Computer Graphics*, 13(5): 925–938.

Wang, P.-S.; Liu, Y.; and Tong, X. 2016. Mesh Denoising via Cascaded Normal Regression. *ACM Trans. Graph.*, 35(6).

Zeng, X.; Vahdat, A.; Williams, F.; Gojcic, Z.; Litany, O.; Fidler, S.; and Kreis, K. 2022. LION: Latent Point Diffusion Models for 3D Shape Generation. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Zhang, W.; Deng, B.; Zhang, J.; Bouaziz, S.; and Liu, L. 2015. Guided Mesh Normal Filtering. *Computer Graphics Forum (Special Issue of Pacific Graphics 2015)*, 34: 23–34.

Zhang, Y.; Shen, G.; Wang, Q.; Qian, Y.; Wei, M.; and Qin, J. 2022. GeoBi-GNN: Geometry-aware Bi-domain Mesh Denoising via Graph Neural Networks. *Computer-Aided Design*, 144: 103154.

Zheng, Y.; Fu, H.; Au, O. K.-C.; and Tai, C.-L. 2011. Bilateral Normal Filtering for Mesh Denoising. *IEEE Transactions on Visualization and Computer Graphics*, 17(10): 1521–1530.