International Journal of Circuit, Computing and Networking

E-ISSN: 2707-5931
P-ISSN: 2707-5923
IJCCN 2020; 1(1): 19-21
Received: 05-05-2020
Accepted: 09-06-2020

Patitapaban Rath
(a) Research Scholar, KIIT University, Bhubaneswar, Odisha, India
(b) Lecturer, Central University of Odisha, Koraput, Odisha, India

Corresponding Author:
Patitapaban Rath
(a) Research Scholar, KIIT University, Bhubaneswar, Odisha, India
(b) Lecturer, Central University of Odisha, Koraput, Odisha, India

Noise removal from 3D Meshes: An efficient approach

Patitapaban Rath

DOI: https://doi.org/10.33545/27075923.2020.v1.i2a.16

Abstract
The current methods for mesh denoising focus on models where all the vertices in the mesh are corrupted with low noise amplitudes. Some of these prevailing techniques would not provide the facility of denoising even when a small percentage of the vertices are corrupted with noise of very high amplitude. To solve the problem, we propose an efficient mesh denoising algorithm for such models in which vertices are corrupted with both high as well as low noise amplitudes. This proposed algorithm is divided into two phases. In the first phase, the corrupted vertices are detected by means of a filter called Localized Coordinate Variation filter. In the second phase, the corrupted vertices are abolished. The results revealed in this paper depicts the claims as mentioned above.

Keywords: mesh denoising; cardinal splines; random valued noise; LCV

1. Introduction

Meshes are basically the aegis for modelling the 3D world. The things responsible for creating noise in meshes are errors in the transmission process, error in the memory location, fault in the hardware or software etc. This incorrect or corrupted information later get in touch with operations like smoothing, posture transformations etc., to be performed on the mesh. The noise having random values transforms the co-ordinates an uncorrupted vertex to an unpredictable value. This value is in the allowed dynamic range for co-ordinates. The main ambition of eradication of noise is to suppress the corrupted information while preserving the other very fine details of the mesh. The Vertex-based Diffusion and VBA (Vertex-based Anisotropic) mesh denoising techniques has been proposed by Zhang *et al*. [1]. Mesh denoising with filtering (MF), denoising with and Linear Heat Diffusion (LHD), Sobolev Regularization (SBR) have been proposed by Peyré in [2]. A piecewise surface denoising method based on consistent sub- neighborhoods was proposed in which claimed to preserve the features of the mesh. He *et al*. proposes a mesh denoising technique via L0 minimization [3], where the flat regions in the noisy mesh are maximized gradually while the sharp features are preserved. Flieshman *et al*. [4], reveal a simple and fast an-isotropic mesh denoising algorithm. Sun *et al*. [5] tabled a feature preserving mesh denoising algorithm by iteratively filtering the face normal at the noisy vertices. Wang *et al*. [6] presented a denoising model based on decoupling local geometric features from the spatial location of a mesh. [8 & 9] proposed efficient and robust image denoising methods using Cardinal Splines [11] and implemented the same to 3D Meshes for high amplitude de-noising. Most of the above revealed techniques proposed for mesh denoising work tremendously with low noise amplitude even when all the vertices in the mesh are corrupted. However, in graphic acquisition, it might be very much possible that the vertices are corrupted with, although the various table text styles are provided. The formatter will need to create these components, incorporating the applicable criteria that follow. Although, it might be very difficult or even next to impossible to design a denoising algorithm for a model in which all the vertices in the mesh are corrupted with very high noise amplitude, it is certainly possible to design a method when a particular percentage of the vertices are corrupted with high amplitude noise. In this paper, the proposed algorithm for 3D de-noising is explained in section 2, performance measurement in section 3, results in section 4 and conclusion in section 5.

2. Proposed algorithm

The proposed algorithm is elaborated in thirteen steps. They are as follows.
Step 1: The variation among the coordinates of the neighboring vertices with respect to the pivot vertex coordinates needs to be calculated first. Hence difference from the pivot vertex of the array with each and every other vertex is calculated.

Step 2: Arrange the differences in ascending order.
Step 3: Take the new variable to store after completion of Step 2. The vertices whose coordinates are closest to the pivot vertex are brought to the initial positions in the set. This operation is performed to have a knowledge of the relative magnitude of the coordinates of the neighboring vertices with respect to the pivot vertex.
Step 4: Variation will be calculated in this step.
Step 5: We choose to calculate the variation between "three" elements in particular because; we need four noise free vertices for cardinal spline interpolation.
Step 6: We are interested in the vertices whose coordinates are least varying.
Step 7: The four least varying coordinates in the neighborhood of the pivot vertex will correspond to the noise free co-ordinates.
Step 8: The four least varying coordinates in the neighbourhood of the pivot vertex will correspond to the noise free co-ordinates.
Step 9: Threshold is to be calculated.
Step 10: The arithmetic mean of the four noise free coordinates is calculated.
Step 11: If the coordinates of the pivot vertex is not within the allowed value from the arithmetic mean of the noise free vertices, then it called as noisy.
Step 12: Basis functions are calculated
Step 13: Once we calculate the basis functions, we interpolate the noise free coordinates and generate the points.

3. Performance measurement

The following parameters are used to check the quality of reconstructed images.

3.1. Peak Signal to Noise Ratio (PSNR)

The proposed algorithms have been tested for different levels of noise ranging from as low as 5% to as high as 95%. The experimental results have been gauged using the mean square error (MSE) and peak signal-to-noise ratio (PSNR) measures that have been given below.

$$MSE = \frac{1}{mn}\sum_{x=0}^{m-1}\sum_{y=0}^{n-1}(A(x,y)-R(x,y))^2$$

$$PSNR = 10\log_{10}\left(\frac{\max^2}{MSE}\right)$$

Where max is the maximum possible pixel value of the image and its value is 255 in the case of a grayscale image. A and R are the original and the restored images having a resolution of m × n.

3.2. Structural Similarity Index Measure – SSIM

The Structural Similarity Index Measure (SSIM) [23] is a full reference metric, in other words, the measuring of image quality based on an initial uncompressed or distortion-free image as reference. SSIM is designed to improve on traditional methods like peak signal to-noise ratio and mean-squared error, which have proved to be inconsistent with human eye perception. The measure of structural similarity compares local patterns of pixel intensities that have been normalized for luminance and contrast. In practice, a single overall index is sufficient enough to evaluate the overall image quality

$$SSIM(x,y) = \frac{(2\mu_x\mu_y - C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)}$$

$$\mathrm{MSSIM}(x,y) = \frac{1}{M}\sum_{n=1}^{M}\mathrm{SSIM}(x_n, y_n)$$

where M and N are total number of n samples
where μ_x is mean and σ_x is Standard Deviation of image x
and μ_y is mean and σ_y is Standard Deviation of image y
and σ_{xy} is covariance of x and y. C_1 and C_2 are constants given below

$$C_1 = (k_1 L)^2 = (0.01\times 255)^2 = 6.5025$$

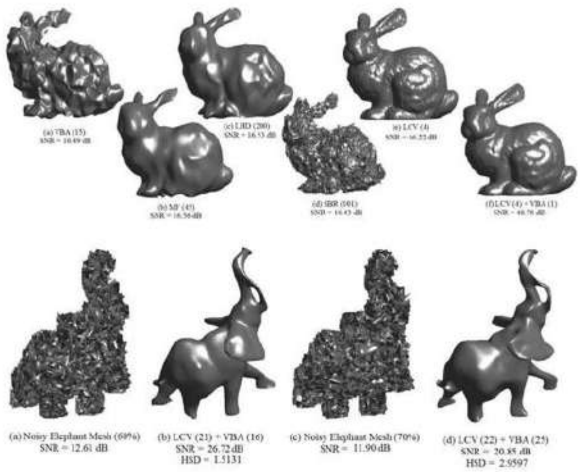
$$and\ C_2 = (k_2 L)^2 = (0.03\times 255)^2 = 58.5225$$

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\sigma_x = [\ \frac{1}{N-1}\sum_{i=1}^{N}(x_i-\mu_x)^2)\ ]^{1/2}$$

$$\sigma_{x,y} = [\ \frac{1}{N-1}\sum_{i=1}^{N}(x_i-\mu_x)(y_i-\mu_y)\ ]^{1/2}$$

4. Results & discussions



5. Conclusion

This paper efficiently reveals the properties of time efficiency and mesh denoising. It is a cumbersome task to implement the features of mesh denoising when all the vertices of a mesh are corrupted with high noise amplitude. But still there is a possibility to reoriginate noisy meshes when few of the vertices are corrupted with high noise amplitude. To eradicate this problem, we propose LCV algorithm which not only eradicates high noise amplitude noise but also low magnitude noise.

6. References

1. Ying Zhang, Ben Hamza A. Vertex-based diffusion for 3-D mesh denoising, IEEE Transactions on Image Processing. 2007; 16(4):1036-1045.
2. Peyré G. The Numerical Tours of Signal Processing - Advanced Computational Signal and Image Processing, IEEE Computing in Science and Engineering. 2011; 13(4):94-97.

3. Lei He, Scott Schaefer. Mesh Denoising via L0 minimization, ACM Transactions on Graphics (TOG). 2013; 32(4):64.
4. Fleishman Shachar, Iddo Drori and Daniel Cohen-Or, "Bilateral Mesh Denoising. ACM Transactions on Graphics. 2003; 22(3):950-953.
5. Xianfung Sun, Paul L. Rosin Ralph R, Martin, Frank C. Langbein, Fast and effective feature-preserving mesh denoising, IEEE Transactions on Visualization and Computer Graphics. 2007; 13(5):925-938.
6. Youyi Zheng, Hongbo Fu, OK-C Au, Chiew-Lan Tai. Bilateral Normal Filtering for Mesh Denoising, IEEE Transactions on Visualization and Computer Graphics. 2011; 17(10):1521-1530.
7. Ruimin Wang, Zhouwang Yang, Ligang Liu, Jiansong Deng, Falai Chen. Decoupling noise and features via weighted ℓ analysis compressing sensing, ACM Transactions on Graphics. 2014; 33:2.
8. Bodduna K, Rajesh Siddavatam. A novel algorithm for detection and removal of random valued impulse noise using cardinal splines, IEEE INDICON, 2012, 1003-1008.
9. Rajesh Siddavatam, Syamala Jayasree, Kireeti Bodduna, Prasant Kumar Pattnaik. An Expeditious cum Efficient Algorithm for Salt-and-Pepper Noise Removal and Edge-Detail Preservation using Cardinal Spline Interpolation, Elesvier Journal of. Vis. Commun. Image Representation. 2014; 25:1349-1365.
10. Michael Unser, Splines: A Perfect Fit for Signal and Image Processing, IEEE Signal Processing Magazine. 1999; 16(6):24-38.
11. Donal Hearn, Pauline Baker M. Computer Graphics withOpen GL, 3rd Edition, Pearson Publishers, 2009.