

# Mesh Smoothing via Adaptive Bilateral Filtering

Qibin Hou<sup>1</sup>, Li Bai<sup>1</sup>, and Yangsheng Wang<sup>2</sup>

<sup>1</sup> School of Computer Science and IT, University of Nottingham,  
Nottingham, NG8 1BB, UK  
{qbh, bai}@cs.nott.ac.uk,

<sup>2</sup> Institute of Automation, Chinese Academy of Sciences,  
Beijing, 100080, China  
wys@nlpr.ia.ac.cn

**Abstract.** In this paper, we present an adaptive bilateral filtering algorithm that can be used to remove unavoidable noise from 3D mesh data generated by initial stages. Selecting the parameters for bilateral filters automatically, this algorithm smoothes meshes in the normal field using anisotropic character of local neighborhood triangles. Experimental results demonstrate that the proposed method remove light noise from meshes and reserve fine features of meshes as good as best results of other methods, with the advantage of none user-assisted parameters setting. Visual comparisons display that the method proposed in this paper performs better than other smoothing method for heavy noisy mesh.

## 1 Introduce

Over recent years, 3D computer graphics research results have been applied in interactive entertainment industry to meet the public's increasing thirst for fun at an enormous rate. In many applications, 3D mesh processing act an important role for visual representation of 3D content as realistic as possible. This depends on the quality of 3D data which is created by scanning or modelling process. However, the raw geometry data acquired from the real-world object usually contains undesirable noise from various sources, even with high fidelity scanners [1]. Therefore, smoothing is necessary before further stages, with the purpose to remove noise while preserving the crucial shape features.

Amount of mesh smoothing or mesh denoising algorithm have been published so far, for example [2], [3], [4], [5], [6], [7], [8], [9], [10]. Taubin [2] put forward a simple, linear and isotropic method to enhance the smoothness of triangulated surfaces at first. After that, Desbrun et al [3] made an improvement by using a geometric flow analogy for irregular meshes. Inspired by image processing work on scale space and anisotropic diffusion, Zhang [7] and Bajaj [10] proposed different feature-preserving surface fairing method. Ohtake [4], Taubin [5], Tasdizen [6] offered diffusion-type smoothing on the normal field itself. Jones [9] analyzed the advantages and disadvantages of all these methods, and achieved impressive results based on robust statistics and local first-order predictors of the surface.

He utilized a method analogous to bilateral filtering for images. Contemporaneously, Fleishman [8] presented a similar method with Jones that uses bilateral filtering for smoothing surfaces, but exploited a different distance way. However, both the methods exploited by Jones [9] and Fleishman [8] are needed to choose parameters assisted by user.

By integrating Ohtake [4] with Fleishman [8] together, this paper attempts to present an idea of developing self-adaptive parameters setting bilateral filter for noisy triangle meshes. The aim is to automatically smooth a noisy mesh and preserve sharp features without help from user. The rest of this paper is organized as follows: Section 2 describes background about bilateral filtering for images, and adaptive Gaussian filter, and how to adapt and extend them for meshes. Section 3 describes our adaptive bilateral filter smoothing method. Experiment results are shown in Section 4. Section 5 has conclusion and proposal for future work.

## 2 Background About Bilateral Filter and Adaptive Gaussian Filter

### 2.1 Bilateral Filter for Images and Its Extension to Meshes

Bilateral filter is proposed by Tomasi et al. [11] for edge-preserving image smoothing. For image  $I$ , the formulation to calculate the output value at pixel  $u(x, y)$  is [11]:

$$E(u) = \frac{\sum_{p \in N(u)} G(\|p - u\|) S(|I(p) - I(u)|) I(p)}{\sum_{p \in N(u)} G(\|p - u\|) S(|I(p) - I(u)|)} \quad (1)$$

where  $N(u)$  is the neighbourhood of  $u$ .  $G(\|p - u\|)$  measures the geometric closeness,  $S(|I(p) - I(u)|)$  measures the photometric similarity. It can be seen that  $E(u)$  depends not only on the spatial distance  $\|p - u\|$ , but also on the signal difference  $|I(p) - I(u)|$ .

In order to filter a mesh triangle  $T$  using local neighboring triangles  $T_j$ , we try to extent Formula (1) for processing mesh normals. The triangle normals can be seemed as vector-valued image defined over the meshes. Let  $n(T)$  is the normal value at triangle  $T$ ,  $C$  is the centroid of  $T$ ,  $n(T_j)$  is the normal value at triangles  $T_j$ , and  $C_j$  is the centroid of  $T_j$ . Spatial weight  $G$  depends on the geodesic distance  $d(C, C_j)$ .  $d(C, C_j)$  is found via Dijkstra's algorithm on the dual mesh which is formed by the centroids [4]. Influence weight  $S$  depends on the difference between  $n(T)$  and  $n(T_j)$ , which is denoted by  $h(T, T_j) = \|n(T) - n(T_j)\|$ .

The smoothed normal  $m(T)$  is computed by

$$m(T) = \frac{\sum A(T_j) G(d) S(h) n(T_j)}{|\sum A(T_j) G(d) S(h) n(T_j)|} \quad (2)$$

where  $A(T_j)$  is the area of  $T_j$ .

According to Jones [9], the spatial weight and the influence weight are able to use Gaussian filter. By selecting width of  $\sigma_g$  of the spatial and  $\sigma_s$  of the influence weight Gaussians, the amount of smoothing is achieved.

Mesh normal of triangle  $T$  due a Bilateral filtering can also be described as:

$$n(T) = m_{\sigma_g, \sigma_s}(T) + \varepsilon_{\sigma_g, \sigma_s}(T) \tag{3}$$

where  $\varepsilon_{\sigma_g, \sigma_s}$  is the difference between original normal and the smoothed normal.

### 2.2 Adaptive Gaussian Filter for Local Smoothness

Gomez [12] gave an adaptive Gaussian filter in which local variance  $\sigma$  for each kernel is selected automatically by working with a scale-space framework and minimal description length criterion (MDL).

The filtering procedure can be understood as [12]:

$$I_0(x, y) = I_\sigma(x, y) + \varepsilon_\sigma(x, y) \tag{4}$$

where  $I_\sigma(x, y)$  is the Low-Pass part, and  $\varepsilon_\sigma(x, y)$  is the residual part. Rewrite Equation (4) as description length (dl), [12] has following equation:

$$dl_{I_0} = \left(\frac{\gamma}{\sigma_x^2}\right) + \varepsilon^2, \gamma = \frac{2n\sigma_\varepsilon^2\alpha\beta}{k} \tag{5}$$

where  $n \geq 2$ ,  $\alpha, \beta$  are given constants.

An image smoothed by adaptive Gaussian filter is [12]:

$$I_\sigma(x, y) = I_0(x, y) \exp\left(-\frac{x^2 + y^2}{2\sigma^*(x, y)^2}\right) \tag{6}$$

where  $\sigma^*(x, y)$  is local variance at selected scale from  $\sigma_1, \sigma_2, \dots, \sigma_n$ . At that scale, description length of  $I_\sigma$  has minimal value at location  $(x, y)$ .

## 3 Adaptive Bilateral Mesh Smoothing

In our proposed method, bilateral filter and adaptive parameter selection are combining together for smoothing the normal fields. And then, fitting meshes to modified normals to achieve smoothed results.

### 3.1 Normal Smoothing Algorithm Description

Define  $\Omega$  is composed by the  $\sigma_{min}, \sigma_{max}$ , and step length  $\lambda$ , these three variances are decided by average edge length of all the meshes. Please refer to Equation (2) for means of other variances, for example,  $n$  is the normal of  $T$ ,  $C$  is the centroid of  $T$ , the neighborhoods of  $T$  denoted by  $\{T_j\}$ , etc. The pseudo-code for applying Bilateral filter to triangle normal with parameter automatic setting is as follows:

**Name:** Adaptive Bilateral Normal Smoothing

**Input:**  $\{T, n, \Omega | T = (v_1, v_2, v_3); n(T) = (x, y, z); \Omega = (\sigma_{min}, \sigma_{max}, \lambda)\}$

**Output:**  $m(T) = (x', y', z')$

**Step 1:** Search for neighborhoods  $\{T_j\}$ , which meet the requirement of  $D(C_j, C) \leq 2\sigma_{max}, j = 1, \dots, k$

**Step 2:** Calculate influence weight parameter and Gaussian filtering

$$S(h) = \exp(-\frac{h^2}{2\sigma_s^2}), \text{ where } \sigma_s = \sqrt{\frac{1}{k} \sum_{j=1}^k (h_j - \bar{h})^2},$$

$$\bar{h} = \frac{1}{k} \sum_{j=1}^k h_j, h_j = \|n(T) - n(T_j)\|$$

**Step 3:** Estimate spatial weight parameter with MDL at scale space

**FOR**  $j = 1 : 1 : k$

**FOR**  $\sigma_i = \sigma_{min} : \lambda : \sigma_{max}$

$$G(d) = \exp(-\frac{d^2}{2\sigma_i^2})$$

$$m_{\sigma_i, \sigma_s}(T) = \frac{\sum A(T_j)G(d)S(h)n(T_j)}{|\sum A(T_j)G(d)S(h)n(T_j)|}$$

$$\varepsilon^2 = 2(1 - |\frac{\sum A(T_j)G(d)S(h)n(T_j)}{\sum A(T_j)G(d)S(h)}|)$$

**END**

**END**

**THEN**  $m_{\sigma_g, \sigma_s}(T) = \{m_{\sigma_i, \sigma_s}(T) | \sigma_g = [\operatorname{argmin}(\frac{\mu}{\sigma_i^2}) + \varepsilon^2]\}$

**Step 4:** Get smoothing result, normalize the average normal  $m(T) = m(T)/\|m(T)\|$

### 3.2 Mesh Fitting Based on Modified Normals

From Sec. 3.1, the smoothed normal for each triangle has been obtained. The further need to do is to rebuild mesh from the smoothed normals. Ohtake [4] gave an efficient method to modify the mesh vertex positions for fitting mesh.

For each mesh vertex  $V$ , [4] define

$$E_{fit}(V) = \sum_{j=1}^n A(T_j) |n(T_j) - m(T_j)|^2 \quad (7)$$

where  $n$  is the amount of all triangles that are adjacent to  $V$ .

Then new position for  $V$  is

$$V_{new} = \operatorname{argmin}(E_{fit}(V)) \quad (8)$$

A simple conjugate gradient method [13] is used to explore the minimum of  $E_{fit}(V)$ . Please refer to [4] for detail implementation.

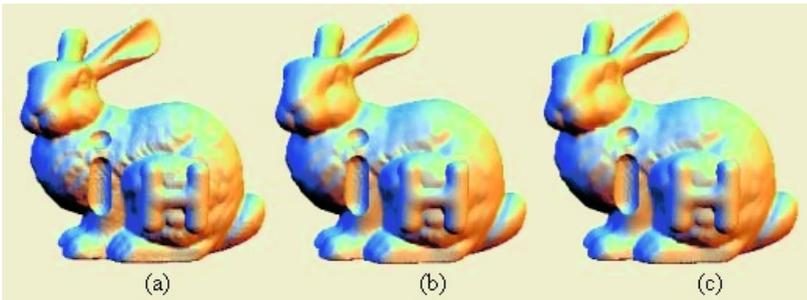
## 4 Results and Comparison

We realized the adaptive bilateral mesh smoothing methods described in last section based on the platform provided by Ohtake. Fig. 1(c) demonstrates our result, comparing with the result in Fig. 1(b) from Ohtake’s adaptive Gaussian and anisotropic neighborhood [4].

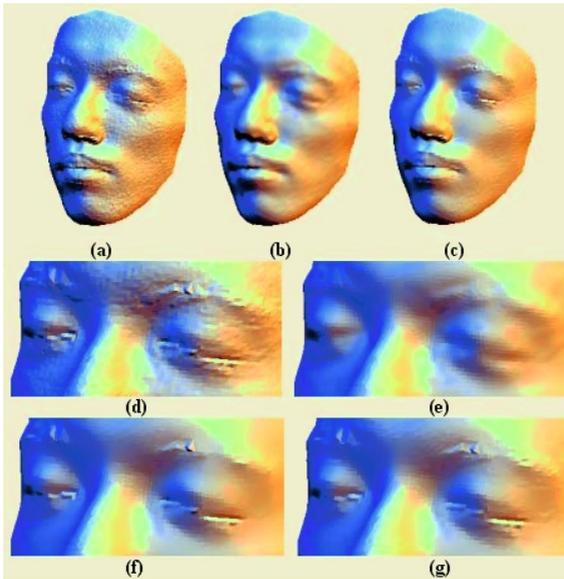
In all cases of this faction, Gaussians are used for spatial ( $G$ ) and influence ( $S$ ) weight with standard deviation of  $\sigma_g$ ,  $\sigma_s$  individually. Other robust weight functions can be applied as well. Jones [9] found Gaussian has good performance in those two weights. In Fig. 1, the choice for  $\sigma_s$  depends on Gaussian error normal of neighborhoods, which were selected with the help of  $\sigma_{max}$ , the maximum value of  $\sigma_g$ . To find  $\sigma_g$ , we use ten scale-spaces with  $\sigma$  from  $\sigma_{min} = 0.4l$  to  $\sigma_{max} = 4.0l$  with step size  $\lambda = 0.4l$ , something like what Ohtake have done in [4].  $l$  is the arithmetic mean of the edge lengths of the mesh. For each mesh triangle, local values of  $\sigma_g$ ,  $\sigma_s$  are set by algorithm adaptively, not by user. From Fig. 1, we can see that our method makes same good smoothing as that best result produced by Ohtake.

It is known that isotropic methods indiscriminately smooth noise and outstanding features at same time [9]. Many researchers have proposed various anisotropic method to overcome this limitation of isotropic smooth technique. Durand and Dorsey [14] show that bilateral filtering is alike with anisotropic diffusion. This important character of bilateral filtering helps it to remove noise while preserving the shape. A comparison of Taubin’s  $\lambda|\mu$  smoothing method [5] with our adaptive bilateral filtering scheme, is presented by Fig. 2.

From Fig. 2, the vision result of Fig. 2(c) seems smoother at general surface of face than Fig. 2(b), while Fig. 2(g) keeps the eyes of model more eminent than Fig. 2(e). It shows that outstanding features such as eyes were kept, while noises were removed efficiently. This demonstrates the effectiveness of our method at feature preservation.



**Fig. 1.** Results of smoothing Stanford Bunny with word ‘iH’ embossed. (a) Original mesh. (b) Best visual result obtained with Ohtake’s adaptive Gaussian + anisotropic neighborhood. (c) Adaptive bilateral smoothing proposed in this paper. The model is courtesy of A. Belyaev

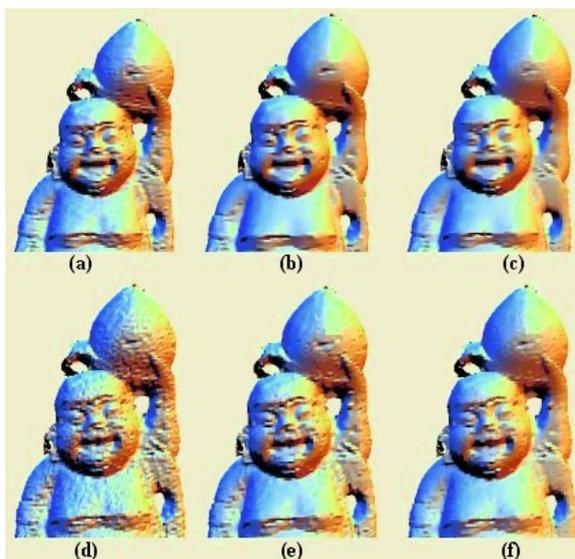


**Fig. 2.** Results of smoothing 3D face model. (a) Original face mesh. (b) Result of Taubin's  $\lambda|\mu$  smoothing method. (c) Result of our method. (d) Zoom the eyes of original model. (e) Zoom the eyes of result from method of Taubin. (f) Zoom the eyes of result from method of Ohtake. (g) Zoom the eyes of result from our method. The model is courtesy of A. Belyaev

Ohtake [4] said that one of drawbacks of his method is not able to remove large noise. We test the capability of our method by smoothing mesh at different noise level in Fig. 3. Fig. 3(a) is a monk mesh obtained using laser scanner. Fig. 3(a) is smoothed by adaptive Gaussian and anisotropic neighborhoods, which produces Fig. 3(b). Fig. 3(c) is our own result. Fig. 3(c) and Fig. 3(b) have similar visual quality. This means that both our method and Ohtake's method are valuable for light noisy mesh. However, when they cope with heavy noisy mesh, there will be a little different effect, as shown in Fig. 3(e) and Fig. 3(f) individually. Fig. 3(d) is obtained by adding heavy noise to Fig. 3(a). Fig. 3(e) is the result of Ohtake's method used to smooth Fig. 3(d). Fig. 3(f) is got by adaptive bilateral filter. A slight improvement of visual quality is achieved in Fig. 3(f) comparing with Fig. 3(e). This result shows the capability of our method to smooth mesh in the presence of little heavy noise.

## 5 Conclusion

In this paper, we propose adaptive bilateral mesh filtering method in which two parameters are set by fully automatic solution according to the shape of local triangle neighborhoods. This improved method adaptively smoothes light noisy mesh and keep the sharp features as well, with ideal human visual effect. In



**Fig. 3.** Results of smoothing at various noise level. (a) Original monk mesh. (b) Smoothed by Ohtake's Gaussian + anisotropic method. (c) Adaptive bilateral filter was applied. (d) Noisy monk mesh. (e) Ohtake's method for noisy mesh. (f) Our method for noisy mesh

addition, it can also work for large noisy mesh with a little better robustness, and outperforms other methods in a way.

This adaptive bilateral mesh smoothing method is based on iteration. So the time-consuming is the drawback of the developed method. To find avenue for fixing this weakness will be a task for future work. Another work need to do is to establish a general quality evaluation rule or metrics for 3D mesh smoothing. That metrics should be consistent with human perception of 3D shape. This is another important theme for future work, as Belyaev has said [15].

**Acknowledge.** Special thanks to Yutaka Ohtake for providing basic platform on which we can implement our algorithm and comparison efficiently. Original meshes in this paper courtesy of A. Belyaev from his homepage. This work was funded by University of Nottingham (Project Code A2RH04). We would like to thank the anonymous reviewers of this paper for their valuable comments.

## References

1. Rusinkiewicz, S., Hall-Holt, O., and Levoy, M.: Real Time 3D Model Acquisition. *ACM Trans. Graphics* 21, 3, 438-446
2. Taubin, G.: A Signal Processing Approach to Fair Surface Design. In *Processings of SIGGRAPH 1995*, 351-358
3. Desbrun, M., Meyer, M., Schroder, P., and Barr, A. H.: Implicit Fairing of Irregular Meshes Using Diffusion and Curvature Flow. In *Proceedings of SIGGRAPH 1999*, 317-324

4. Ohtake, Y., Belyaev, A. G., and Seidel, H. P.: Mesh Smoothing by Adaptive and Anisotropic Gaussian Filter. In *Vision, Modeling, and Visualization 2002*, 203-210, Erlangen, Germany, November 2002.
5. Taubin, G.: Linear Anisotropic Mesh Filtering. IBM Research Report RC22213 (W0110-051), IBM, October 2001
6. Tasdizen, T., Whitaker, R., Burchard, P., and Osher, S.: Geometric Surface Smoothing via Anisotropic Diffusion of Normals. In *Proceedings, IEEE Visualization 2002*, 125-132
7. Zhang, H., and Fiume, E. L.: Mesh Smoothing with Shape or Feature Preservation. In *Advances in Modeling, Animation, and Rendering*, J. Vince and R. Earnshaw, editors, 167-182, 2002
8. Fleishman, S., Drori, I., and Cohen-Or, D.: Bilateral Mesh Denoising. *Proceedings of ACM SIGGRAPH 2003*
9. Jones, T., Durand, F., and Desbrun, M.: Non-iterative, Feature-preserving Mesh Smoothing. *ACM Trans. Graphics* 2003, 22, 3, 943-949
10. Bajaj, C., and Xu, G.: Anisotropic Diffusion on Surfaces and Functions on Surfaces. *ACM Trans. Graphics* 2003, 22, 1, 4-32.
11. Tomasi, C., and Manduchi, R.: Bilateral Filtering for Gray and Color Images. In *Proc. IEEE Int. Conf. On Computer Vision*, 1998, 836-846
12. Gomez, G.: Local Smoothness in Terms of Variance: the Adaptive Gaussian Filter. In *Proc. Of the BMVC 2000*, vol.2, 815-824
13. Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P.: *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1993.
14. Durand, F., and Norsey, J.: Fast Bilateral Filtering for the Display of High-Dynamic-Range Images. *ACM Trans. Graphics* 2002, 21, 3, 257-266
15. Belyaev, A., and Ohtake, Y.: A Comparison of Mesh Smoothing Methods. *Israel-Korea Bi-National Conference on Geometric Modeling and Computer Graphics* Tel-Aviv, Feb. 2003, 83-87