Curve and Surface Smoothing without Shrinkage

Gabriel Taubin

IBM T.J.Watson Research Center
P.O.Box 704, Yorktown Heights, NY 10598

IGR202

Professors:
Amal DEV
Kiwon UM

Student:
Giulia MANNAIOLI

Contents

1 Introduction

This paper discusses the problem of faceting in approximated shapes and introduces a new linear low-pass filter method for smoothing piece-wise linear shapes of any dimension and topology without producing shrinkage. Previous smoothing methods, such as Fourier descriptors and Gaussian smoothing, suffer from computational complexity or shrinkage issues. The paper shows how to define Gaussian smoothing on polyhedral surfaces of arbitrary topology, but also points out its shrinkage problem. The new method presented in this paper, the Taubin method, has linear computational complexity and storage requirements and can be applied to general piece-wise linear shapes. The analysis techniques presented in the paper generalize Lindeberg's analysis to any dimension and topology. The algorithm replaces each vertex's position with a convex combination of its position and its neighbors. The new smoothing algorithm consists of two consecutive passes of Gaussian (but actually it presents the Laplacian filter) smoothing with alternating positive and negative scale factors. It produces a low-pass filter effect, where the curvature of the curve or surface replaces the frequency. The original curve or surface is modeled as a smooth curve or surface underlying a normal perturbation vector. The quantity of attenuation is determined by the number of iterations N. The filtering design requires determining the first and second scale factors lambda and mu and the number of iterations N based on the low-pass filter parameters.

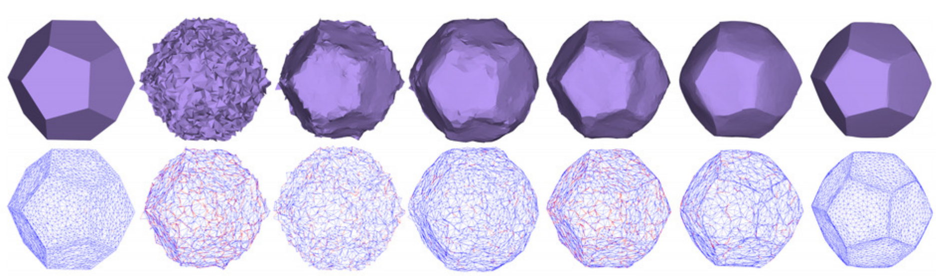
2 Related works

Computer graphics has long studied various smoothing techniques for piece-wise linear shapes, including Fourier descriptors and Gaussian smoothing. However, these methods suffer from computational complexity or shrinkage issues. To address these limitations, other methods have been proposed, such as bilateral mesh denoising, surface denoising, implicit fairing of irregular meshes, Laplace-Beltrami operators, and geometric flow-based methods.

Fleishman et al. introduced bilateral mesh denoising, which can be used for smoothing surfaces, but requires an initial mesh and has a high computational complexity. The paper "Shape Smoothing via Iterative Fairing" proposes a smoothing technique based on the concept of fairing, which approximates the surface of a shape with a continuous and regular surface that minimizes the total energy of the surface itself. It uses a combination of local and global optimization techniques to achieve a regular and stable solution while preserving the structural and geometric features of the original shape.

Similar to the method proposed in this paper, the paper "Geometry-based Intrinsic Polishing for Triangular Meshes" by Kobbelt et al. also utilizes a low-pass filter for mesh smoothing, but uses a bilateral filter that preserves edges.

There are also recent methods using machine learning approaches such as "Normalfnet: Normal filtering neural network for feature-preserving mesh denoising".



3 Technical details

Gaussian smoothing is a popular technique for geometric smoothing of parametric curves, associated with the theory of scale-space. In the continuous case, Gaussian smoothing is performed by convolving the vector function that parameterizes the curve with a Gaussian kernel, which attenuates all frequencies except for the zero frequency. This cause of course a shrinkage problem. To avoid contraction, the smoothing algorithm must produce a low-pass filter effect: the Taubin smoothing. This is an isotropic method used for mesh denoising that aims to smooth the surface while preserving mesh features.

The method used in this implementation is based on the Laplacian operator, and it calculates the new position for each vertex by averaging the neighboring vertices. The algorithm works by iterating over all the vertices of the mesh and moving each vertex towards the average of its neighbors, weighted by a factor . The new position of the vertex is calculated by adding the weighted average of the neighboring vertices to the current position of the vertex. After this step, the algorithm computes a Laplacian of the new mesh, and then moves the vertices again towards the average of their neighbors, but this time weighted by a factor . The new position of the vertex is calculated by subtracting the weighted Laplacian of the vertex from the current position of the vertex. Setting lambda equal to mu in magnitude results in the bilaplacian flow, which is a discrete version of the steepest descent flow for the thin-plate energy functional. This flow can be derived from the Taubin smoothing scheme when the positive and negative scale factors are of equal magnitude. However, the bilaplacian flow does not improve the low-frequency surface features.

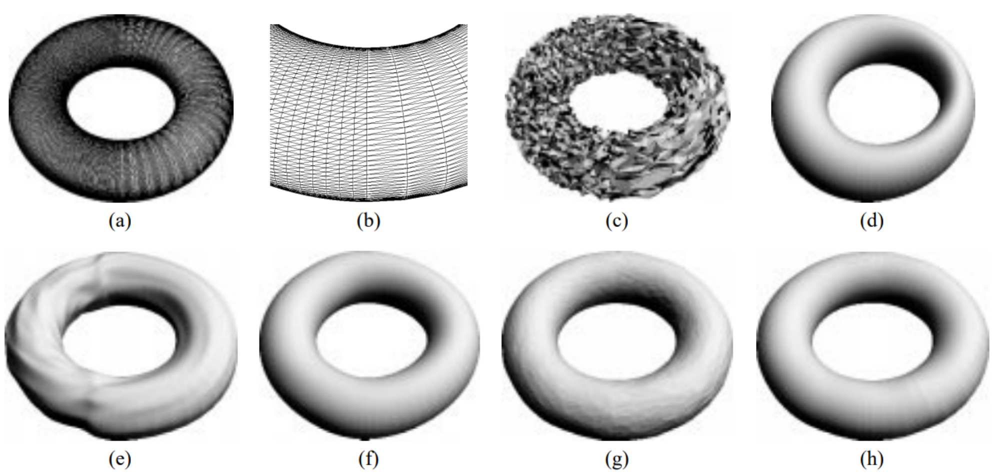
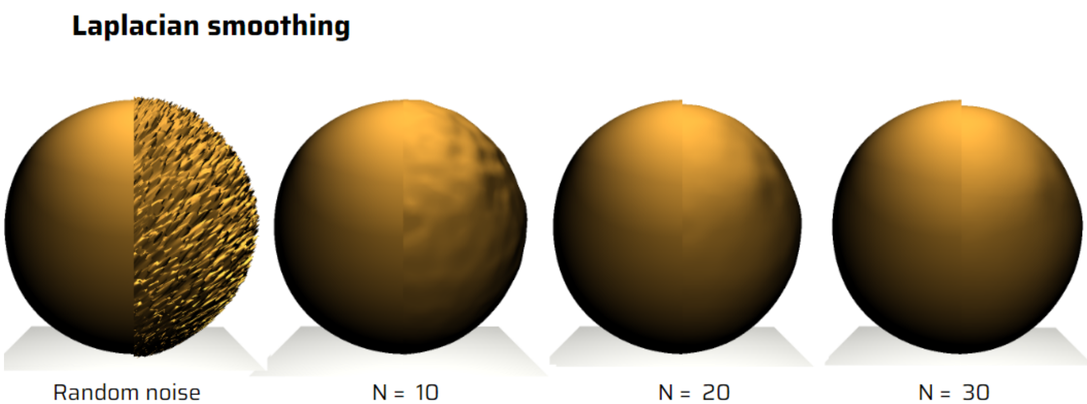
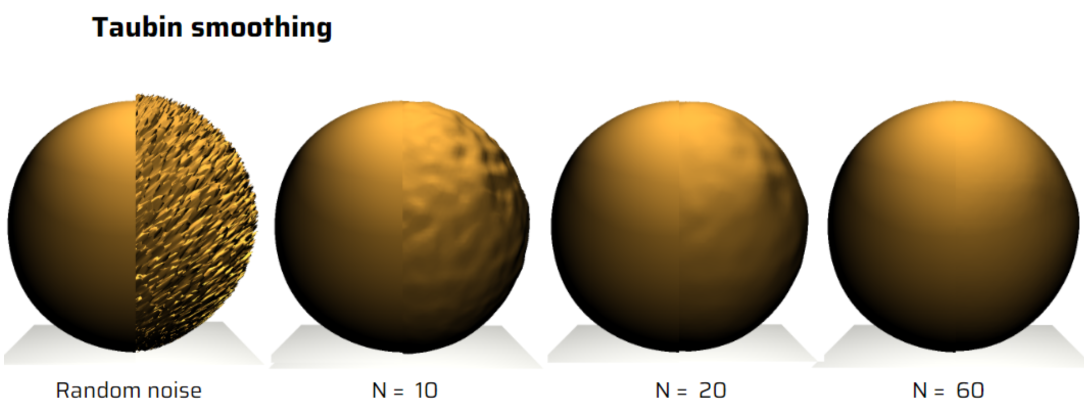


Figure 4. (a) A torus with consisting of two parts with different sampling rates; (b) a magnified view of a part of the torus; (c) the torus with a uniform noise added; (d) Laplacian smoothing deforms the initial shape; (e) smoothing by the Taubin method reduces high-frequency surface oscillations but develops low-frequency surface waves; (f) the bilaplacian flow smoothes well but slightly deforms the initial shape; (g) the mean curvature flow smoothes well but produces irregular mesh; (h) smoothing according to (11).

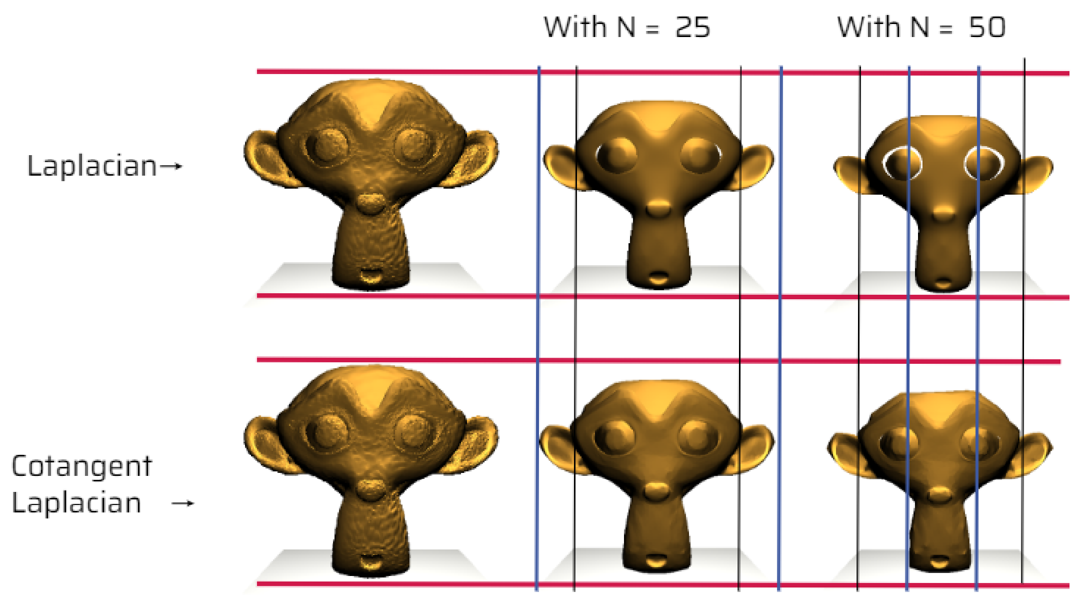
As can be seen from the figures below, the Laplacian filter is much faster but causes shrinkage. Taubin smoothing partially resolves this problem but is much slower. This is due to the fact that it requires many more iterations because after each smoothing, the mesh needs to be re-expanded. Of course it also takes more time for each iteration as there are two filters instead of one. The two filters were made with the same lambda, so if we look at the sphere after the 10th iteration of Laplacian, it appears to work better than Taubin, which after 10 iterations has smoothed but not very well. However, as we continue with the iterations, shrinkage becomes noticeable if we apply only a Laplacian filter.

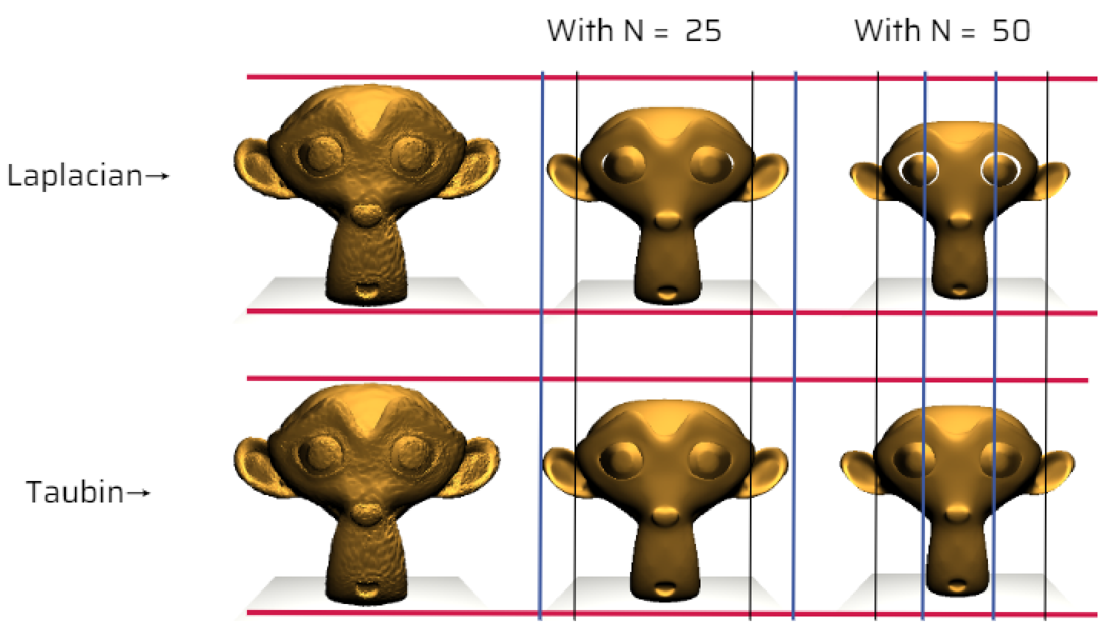




While the results with the sphere are not very clear, the monkey figure shows the difference more clearly.

After aligning the figures based on the floor, it is immediately noticeable that the Laplacian filter causes significant shrinking of the figure, particularly in the mouth area and it is noticeable that there is much more space between the ears and the lines then when using the Taubin smoothing method. This difference becomes even more apparent in the third image. During the implementation, the cotangent Laplacian operator was also tested and it produced significantly better results in terms of preventing shrinking in the mesh, as compared to the standard Laplacian operator. However, it should be noted that the cotangent Laplacian operator was found to be computationally more expensive than the other smoothing methods and that the noise is not perfectly removed.





In the following images, it is better observable the difference between the three methods. The Uniform Laplacian method has effectively removed the noise but has significantly reduced the image, particularly noticeable in the ears. The Cotangent Laplacian reduced the shrinkage effect a lot but does not remove all the noise. The Taubin method is slightly less effective than the uniform laplacian but it has not reduced the mesh at all. However, it required a much higher number of iterations.

4 Modifications and improvements

The article focused on the Gaussian filter, however, upon further research and review of slides, it results that the method described is actually the Laplacian filter.

Gaussian smoothing uses a Gaussian-type filter, which has a symmetric and continuous weighting function. This filter calculates the weighted average of neighboring points for each point in the mesh, with greater weight given to closer points. The effect of this filter is to make the mesh smoother and less detailed, reducing irregularities and small imperfections.

Laplacian smoothing, on the other hand, uses a Laplacian-type filter, which has an asymmetric and discontinuous weighting function. This filter calculates the difference between the position of the mesh point and the average of its neighbors, and then updates the position of the mesh point based on this difference and on a weight that is the same for all the neighbours points. The effect of this filter is to move each point of the mesh towards the average of its neighbors, eliminating small irregularities but preserving the main details of the mesh.

In general, Gaussian smoothing is better suited for removing uniform and subtle noise, while Laplacian smoothing is better suited for removing irregular and coarser noise. In this specific case where the noise is applied randomly to each vertex position, the resulting noise pattern is irregular. As a result, Laplacian smoothing might work better as it is better suited for removing irregular and coarse noise. Gaussian smoothing, on the other hand, is more effective at removing uniform and subtle noise. As in the slides and other examples, the Laplacian filter was therefore used.

During the development of the project, also an attempt was made to use the cotangent function for curve and surface smoothing. However, several issues were encountered most of all to handle the negative value of the weights. Also understanding the final formula to give the new value to the vertex was challenging. Only after several attempts it was possible to resolve all the problems and to apply the cotangent Laplacian. This approach was found to be successful in achieving the desired level of smoothing while significantly reducing mesh deformations and shrinkage. Additionally, the modified approach required fewer iterations (even if the complexity of each iteration is bigger) and maintained the shape of the mesh much better than the Uniform Laplacian.

In contrast to the cotangent-based approach, which relies on the uniform Laplacian, the modified approach, the Taubin method, allows for greater flexibility in preserving the shape of the mesh. As a result, the overall quality of curve and surface smoothing in the project was improved.

Therefore, it can be concluded that the Taubin method was successful in achieving the desired results, and it was a better choice than using cotangent for curve and surface smoothing in this project.

Additionally, due to the heavy dependence of this method on the parameters and the number of required iterations, it was difficult to make an informed decision. As computation time was also a limiting factor, it was challenging to reach a conclusive decision, even after implementing the function. In fact, an attempt was made to determine the best parameters (lambda and mu) by implementing a function that calculates the minimum squared error while ensuring that the constraints are respected. The results of this function were then compared using various parameters.

However, it was not easy to reach a conclusive desicion because the optimal parameter values were also influenced by other factors, such as the number of subdivisions and the amount of noise applied to the model each time.

5 Observations and future works

Firstly, the Laplacian filter can cause a loss of detail in regions of the mesh with high curvature, but this depends on the choice of weights. For example, if there are triangles with a very large angle near a large region of the mesh, then the Laplacian filter may have a greater impact on that region, causing a loss of detail.

Secondly, isotropic methods, such as the Taubin's method, can effectively preserve the geometric features of the mesh, but they may not completely remove the noise due to their global nature. In general, isotropic smoothing is a simpler and faster approach to denoising, but it may result in a loss of information about the orientation of details in the image. On the other hand, anisotropic smoothing refers to a smoothing technique where filtering is selectively applied in different directions in space, taking into account the direction of intensity gradients in the image. In other words, this method applies a stronger filter in regions of the image with a high intensity gradient, while applying a weaker filter in regions with a low intensity gradient. This type of smoothing is often used to preserve the edges of the image and remove only non-directional noise. Therefore, combining isotropic and anisotropic denoising methods may offer the most effective approach to noise reduction, as it takes the advantages of both approaches while minimizing their drawbacks.

Actually, in the Taubin smoothing, there are some limitations to this method that have been identified. Firstly, as we said, it can lead to a loss of detail in the mesh. This is because the smoothing process averages out the vertices, which can result in a loss of high-frequency details (actually is a low-pass filter). Researchers have suggested that one way to mitigate this issue is to use a multiscale approach, where the smoothing is performed at multiple levels of detail.

Secondly, Taubin smoothing can create new artifacts in the mesh, particularly in regions of high curvature. This is because the smoothing process can sometimes cause the mesh to fold in on itself, leading to self-intersections and other issues. Using a curvature-based approach can be useful to avoid these problems, where the smoothing is only applied in regions of low curvature.

Finally, using machine learning techniques could be obviously important to improve Taubin smoothing. By training a machine learning model to learn the characteristics of 3D meshes, it may be possible to develop a more effective smoothing method that can preserve details while also reducing noise. Additionally, there is still place for improvement in the determination of the appropriate scaling factor for isotropic rescaling, and further research in this area could lead to more efficient and accurate denoising methods.

In conclusion, while Taubin smoothing is a useful method for reducing noise in 3D meshes, there are still some limitations that need to be addressed. By using a multiscale approach, a curvature-based approach, combining isotropic and anisotropic denoising methods or machine learning techniques, we may be able to develop more effective and efficient methods for smoothing 3D meshes.

References

[1] Gabriel Taubin. 1995. A signal processing approach to fair surface design. In Proceedings of the 22nd annual conference on Computer graphics and interactive techniques (SIGGRAPH '95). Association for Computing Machinery, New York, NY, USA, 351–358. https://doi.org/10.1145/218380.218473

[2] 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).

[3] Z. Li, Y. Zhang, Y. Feng, X. Xie, Q. Wang, M. Wei, and P.-A. Heng, "Normalf-net: Normal filtering neural network for feature-preserving mesh denoising," ComputerAided Design, p. 102861, 2020.

[4] Shachar Fleishman, Iddo Drori, and Daniel Cohen-Or. 2003. Bilateral mesh denoising. In ACM SIGGRAPH 2003 Papers (SIGGRAPH '03). Association for Computing Machinery, New York, NY, USA, 950–953. https://doi.org/10.1145/1201775.882368

[5] Surazhsky, T., Magid, E., Soldea, O., Elber, G., Rivlin, E. (2003). A comparison of Gaussian and mean curvatures estimation methods on triangular meshes. Center for Graphics and Geometric Computing, Technion, Israel Institute of Technology, Haifa 32000, Israel.