

Multiresolution Analysis for Irregular Meshes

Roy Michael, Sebti Foufou, Andreas Koschan, Frederic Truchetet, Mongi Abidi

▶ **To cite this version:**

Roy Michael, Sebti Foufou, Andreas Koschan, Frederic Truchetet, Mongi Abidi. Multiresolution Analysis for Irregular Meshes. SPIE International Symposium on Photonics Technologies for Robotics, Automation, and Manufacturing, Oct 2003, Providence, RI, United States. hal-01943966

HAL Id: hal-01943966
https://hal.science/hal-01943966

Submitted on 4 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multiresolution Analysis for Irregular Meshes

Michaël Roy[1,2,*], Sebti Foufou[1], Andreas Koschan[2], Frédéric Truchetet[1], and Mongi Abidi[2]

[1] Le2i - CNRS - Université de Bourgogne - 12 rue de la fonderie - 71200 Le Creusot, France
[2] IRIS Lab. - University of Tennessee - 334 Ferris Hall - Knoxville, TN 37996, USA

ABSTRACT

The concept of multiresolution analysis applied to irregular meshes has become more and more important. Previous contributions proposed a variety of methods using simplification and/or subdivision algorithms to build a mesh pyramid. In this paper, we propose a multiresolution analysis framework for irregular meshes with attributes. Our framework is based on simplification and subdivision algorithms to build a mesh pyramid. We introduce a surface relaxation operator that allows to build a non-uniform subdivision for a low computational cost. Furthermore, we generalize the relaxation operator to attributes such as color, texture, temperature, etc. The attribute analysis gives more information on the analysed models allowing more complete processing. We show the efficiency of our framework through a number of applications including filtering, denoising and adaptive simplification.

Keywords: Irregular mesh, multiresolution analysis, surface relaxation, surface attributes

1. INTRODUCTION

3D scanners usually produce large amount of raw data containing geometrical data and attributes. Geometrical data describe shape and dimensions of the object and include data relative to a point set on the object surface. Attributes describe object surface properties such as colors, texture coordinates, temperature, etc. The goal of this work is to build a mesh multiresolution analysis that is capable of managing geometrical data and attributes. The attribute management is very important especially for terrain models where the attributes are linked to the nature of the terrain. In some cases, attributes are more important than the terrain itself. For example, weather data sets contain several attributes (*e.g.* temperature, pressure, humidity, precipitation, etc.) in addition to the terrain (*i.e.* latitude, longitude and elevation of the weather stations recording the data). User-friendly visualization of these kinds of data sets requires management of the attributes.[1]

Multiresolution analysis[2] is an efficient framework to represent a data set at different levels of resolution. The analysis decomposes an initial data set into a sequence of approximations and details. The approximations represent the initial data at different levels of resolution. The details encode the data lost by the approximations and can be seen as a frequency spectrum. The main advantage of the multiresolution analysis is its representational and computational efficicency. Multiresolution analysis is a versatile tool to represent general functions and data sets, and gives rise to many applications such as filtering, denoising, compression, editing, etc.

We propose a multiresolution analysis framework for irregular meshes containing multiple attributes (such as colors, texture, curvature, *etc.*). This framework is based on a popular simplification method called progressive mesh. We use a global downsampling method in order to create disjoint levels of resolution. A fast, local surface relaxation operator is introduced to build a non-uniform subdivision. We show different applications of the multiresolution analysis such as filtering, denoising and simplification of complex models.

In section 1.1, we review mesh multiresolution analysis schemes proposed in the literature. In section 2, we present a new surface relaxation operator and its extension to attribute relaxation. In section 3, we detail the multiresolution analysis scheme. Section 4 presents some experimental results. Conclusion and ideas for future works are given in section 5.

* E-mail: m.roy@iutlecreusot.u-bourgogne.fr, Telephone: +33-385-731-090, Fax: +33-385-731-097

1.1. Related Work

Multiresolution analyses based on wavelet theory have started a new research domain on hierarchical methods for computer graphics.[3] Lounsbery[4] made the connection between wavelets and subdivision to define different levels of resolution. This technique called Subdivision Wavelet Transform makes use of the theory of the multiresolution analysis and of the subdivision rules to construct a multiresolution surface representation for surface with subdivision connectivity. Zorin et al.[5] proposed a combination of subdivision and smoothing algorithms to construct a set of algorithms for interactive multiresolution editing of complex meshes with arbitrary topology. The authors used Loop subdivision for the estimation of the high resolution mesh from the coarse representation.

Those methods assume that the mesh is semi-regular and use traditional subdivision schemes. More recently, different approaches have been presented in order to deal with irregular meshes. Bonneau[6] introduced the concept of multiresolution analysis over non-nested spaces, which are generated by the so-called BLaC-wavelets, a combination of the Haar function with the linear B-Spline function. This concept was then used to contruct a multiresolution analysis over irregular meshes. Kobbelt et al.[7] proposed a multiresolution editing tool for irregular meshes using the discrete fairing method. The authors use the progressive mesh algorithm[8] to build the coarse resolution mesh. A smoothing operator is used to estimate the high resolution mesh. Guskov et al.[9] presented a series of non-uniform signal processing algorithms designed for irregular triangulation. They used a smoothing algorithm combined with existing hierarchical methods to build subdivision, pyramid, and wavelet algorithms for irregular connectivity meshes. The authors proposed a non-uniform subdivision to build a geometrically smooth mesh with the same connectivity as the original mesh. This subdivision is part of the multiresolution decomposition and its central ingredient is a non-uniform relaxation operator.

1.2. Contributions

In this paper, we present a multiresolution analysis framework for irregular meshes containing attributes. Our framework build a mesh pyramid using a simplification and a non-uniform subdivision algorithm. We make the following contributions:

- We present a new surface relaxation operator used for geometric smoothing of triangle meshes. Our scheme is fast, local, and straigtforward to implement.
- We generalize this relaxation operator to surface attributes to build a complete multiresolution analysis.
- We show how to use the multiresolution analysis for irregular meshes in applications such as filtering, smoothing, and adaptive simplification.

2. SURFACE RELAXATION

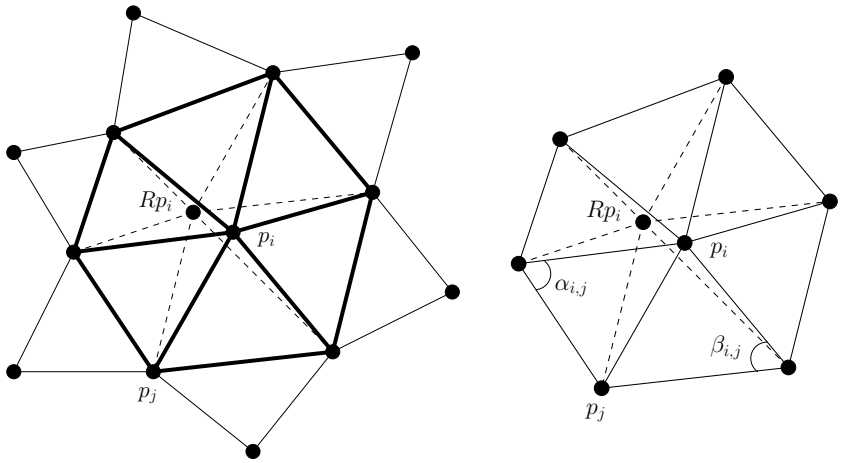
Surface relaxation moves the vertices of a mesh in order to minimize an energy function. Surface relaxation has been heavily applied for surface smoothing using a diffusion process.[10–12] The relaxation operator minimizes the curvature of the mesh and generates a smooth mesh. We use a relaxation operator to build a non-uniform subdivision scheme and create smooth estimation of a fine mesh.

2.1. Geometric Relaxation

Guskov et al.[9] proposed a non-uniform relaxation operator which minimizes second order differences D_e^2 defined at every edge e in the mesh.[13] Later we refer to this method as the *SOD relaxation*. Considering the position $p_i = \{x, y, z\}$ of a vertex v_i, the relaxed position Rp_i of v_i is chosen to minimize the sum of the squares of the second order differences within the support of the vertex v_i:

$$Rp_i = \sum_{j \in \mathcal{V}_2(i)} \lambda_{i,j}.p_j \qquad \text{with} \qquad \lambda_{i,j} = -\frac{\sum_{e,j} c_{e,i}.c_{e,j}}{\sum_e c_{e,i}^2}, \tag{1}$$

where $\mathcal{V}_2(i)$ represents the 1-ring neighborhood with flaps of the vertex v_i (see Fig. 1). The $\lambda_{i,j}$ are weights of the relaxation operator minimizing the second order differences D_e^2 for an edge e. The $c_{e,i}$ are a set of coefficients depending



(a) SOD relaxation (b) Curvature relaxation

Figure 1. Relaxation of a vertex p_i. SOD relaxation requires an extended neighborhood and a local parametrization for every edge of the 1-ring neighborhood (the bold edges). Curvature relaxation is computed according to the angles opposite to the edge $e_{i,j}$ in the 1-ring neighborhood.

	buddha	bunny	dragon	horse
Vertices	543,652	35,947	437,645	48,485
Faces	1,087,716	69,451	871,414	96,966
SOD relaxation	21.70s	1.49s	17.14s	1.91s
Curvature relaxation	5.02s	0.40s	3.95s	0.41s

Table 1. Computation time for SOD relaxation and curvature relaxation operators applied to different models from Stanford University.

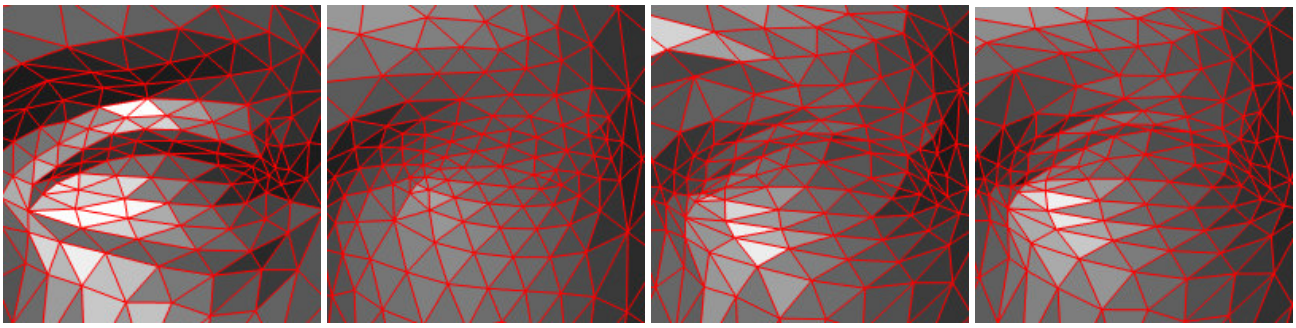
on the geometric position of the mesh vertices.[9] The computation of the coefficients $c_{e,i}$ requires a local parametrization, known as a hinge map, for every edge in the local neighborhood.

Meyer et al.[14] proposed local and accurate discrete differential-geometry operators. One of these operators was previously used for surface denoising.[12,15] It has been shown[16] that this operator gives similar smoothing results to the SOD relaxation operator presented before. The advantage of these two methods is that they smooth the geometry and do not affect the triangle shapes much. We propose a relaxation operator based on surface curvature minimization. We build our non-uniform relaxation operator using the Meyer differential operator. We refer to the following method as the *curvature relaxation*. The relaxed position Rp_i of a vertex v_i is given by:

$$Rp_i = \sum_{j \in \mathcal{V}_1(i)} \lambda_{i,j}.p_j \qquad \text{with} \qquad \lambda_{i,j} = \frac{\cot \alpha_{i,j} + \cot \beta_{i,j}}{\sum_{l \in \mathcal{V}_1(i)} \cot \alpha_{i,l} + \cot \beta_{i,l}}, \tag{2}$$

where $\mathcal{V}_1(i)$ represents the 1-ring neighborhood. The $\lambda_{i,j}$ are weights of the relaxation operator minimizing the curvature energy of an edge $e_{i,j}$ where $\alpha_{i,j}$ and $\beta_{i,j}$ are the angles opposite to the edge $e_{i,j}$ (see Fig. 1).

SOD relaxation operator requires to compute one function for every vertex in the 2-ring neighborhood, and to parametrize every edge in the 1-ring neighborhood. It can be shown that the complexity of the SOD relaxation operator is $O(4n)$ where n is the number of vertices in the 1-ring neighborhood. Curvature relaxation operator requires to compute only one function for every vertex in the 1-ring neighborhood. Therefore, it can be shown that the complexity of the curvature relaxation operator is $O(n)$. Table 1 shows the computation time for the SOD relaxation and the curvature relaxation operators applied to different models. This experiment was performed on a PC with a Pentium 4 at 2.4GHz and with 512MB of memory. The SOD relaxation operator works at about 25,000 vertices per second, while the curvature relaxation operator works at about 100,000 vertices per second. SOD relaxation operator is slowed down by the extended neighborhood requirement and the local parametrization computed for each edge in the local neighborhood.



(a) Initial mesh (b) Laplacian relaxation (c) SOD relaxation (d) Curvature relaxation

Figure 2. Smoothing of the eye of the mannequin (a) using different relaxation operators. The semi-uniform scheme (b) smoothes the surface but distorts the geometry. Non-uniform schemes (c)-(d) only smooth the geometry and preserve triangle shapes.



(a) Initial mesh (b) 1 iteration (c) 20 iterations

Figure 3. Color relaxation of the swirl model. This model is part of a regular sphere. The relaxed colors are nicely smoothed along the surface.

Figure 2 shows the effect of surface relaxation on a non-planar triangulation such as the eye of the mannequin head (Fig. 2(a)). Semi-uniform schemes like the discrete Laplacian (Fig. 2(b)) relaxe a vertex by replacing the vertex with the average of its 1-ring neighbors. The semi-uniform scheme tries to make edge lengths as uniform as possible, which leads to undesired tangential drifts, generating triangles of similar sizes. Non-uniform shemes (Figs. 2(c)-2(d)) smooth the geometry without affecting the triangle shapes much. We see that the two methods shown gives similar visual results.

2.2. Attribute Relaxation

Meshes coming from real world scenes usually contain attributes such as colors, texture coordinates, temperature, radiation, etc. We assume that the attributes are intrisincally linked to the surface. We want to extend the previous relaxation operator to include the attributes. The basic idea is to relax the attributes according the surface geometry.

A vertex v is now represented as a set $(a_1, \ldots, a_N)$ of N attributes a_n. We define an application $f_n(v) = a_n$ that gives the n^{th} attribute a_n associated with the vertex v. The geometrical position of the vertex is then simply defined as an attribute of that vertex. We generalize the previous relaxation operator to an attribute a_n as follows:

$$Rf_n(v_i) = \sum_{j \in \mathcal{V}_1(i)} \lambda_{i,j}.f_n(v_j), \tag{3}$$

where the $\lambda_{i,j}$ are defined as in Equation 2, and only depends on the geometry of the surface. Predicted attributes are relaxed in terms of curvature energy of the analysed surface. Thus, this relaxation operator guaranties smooth attribute variation.

Figure 3 shows the relaxation of color attributes on the swirl model* (Fig. 3(a)). This model is a part of a regular sphere, so the curvature is constant and so are the $\lambda_{i,j}$ coefficients of the relaxation formula. Figure 3(b) shows the result after 1 iteration of the relaxation operator, and Fig. 3(c) shows the result after 20 iterations. We see that the color values are uniformly smoothed following the surface curvature.

3. MULTIRESOLUTION ANALYSIS

Multiresolution analysis provides a framework that rigorously defines various approximations and fast analysis algorithms. This framework decomposes an original data set into a sequence of levels of resolution. We use a decomposition scheme based on the Burt-Adelson pyramid[17] and allows the creation of mesh pyramid.[9] We start from the finest mesh $\mathcal{M}^0$ and compute a sequence of meshes $\mathcal{M}^k$ $(1 \leq k \leq K)$ as well as oversampled differences $\mathcal{D}^k$ between meshes. A level of resolution is defined by a couple $(\mathcal{M}^k, \mathcal{D}^k)$, except for the finest level which is $\mathcal{M}^0$. To go from $\mathcal{M}^k$ to $\mathcal{M}^{k+1}$, we follow the diagram in Fig. 4.

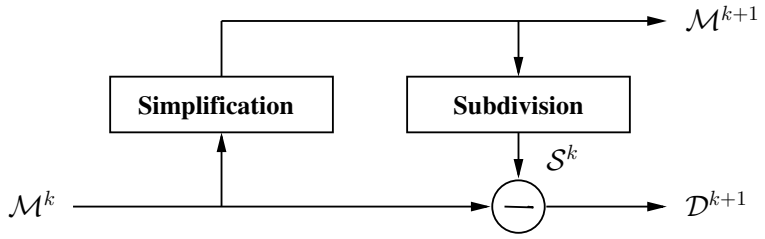


Figure 4. Mesh multiresolution analysis scheme using Burt-Adelson like pyramid.

The decomposition of one level of resolution is decomposed in three steps:

- **Simplification.** The simplification *downsamples* the initial mesh $\mathcal{M}^k$ by removing some vertices, giving the coarse mesh $\mathcal{M}^{k+1}$. The coarse mesh $\mathcal{M}^{k+1}$ is an approximation of the initial fine mesh $\mathcal{M}^k$.
- **Subdivision.** The subdivision *upsamples* the coarse mesh $\mathcal{M}^{k+1}$ by inserting the previously removed vertices, and *relaxes* all the vertices, giving an estimation $\mathcal{S}^k$ of the fine mesh.
- **Detail computation.** The *difference* between the fine mesh $\mathcal{M}^k$ and the subdivided mesh $\mathcal{S}^k$ gives the detail coefficients $\mathcal{D}^{k+1}$ associated with the mesh $\mathcal{M}^{k+1}$. The details are computed for every vertex of the fine mesh $\mathcal{M}^k$.

To construct the mesh pyramid, we need downsampling, upsampling, and relaxation operators. The reconstruction scheme is simply obtained by inverting the decomposition scheme. We start with a coarse mesh $\mathcal{M}^{k+1}$, subdivide it and add the details $\mathcal{D}^{k+1}$ in order to recover the fine mesh $\mathcal{M}^k$.

In this section, we explain the global downsampling and upsampling methods used respectively in the simplification and in the subdivision. Then we detail the decomposition algorithm that creates the levels of resolution, and the reconstruction algorithm used to recover the initial fine data set.

3.1. Global Downsampling and Upsampling

The decomposition is based on an incremental mesh decimation, where the basic step consists of removing one vertex and of re-triangulating the remaining hole. Since our decomposition should be invertible, however, we have to choose a decimation method that is also invertible. An invertible mesh decimation method was introduced in the Progressive Mesh (PM) framework.[8] In the PM setting, a sequence of edge collapses simplifies the mesh, while a sequence of vertex splits recovers the initial mesh. In our framework, an half-edge collapse provides the atomic downsampling step, and a

*http://graphics.cs.uiuc.edu/∼garland/research/quadrics.html

vertex split becomes the atomic upsampling step. We prefer half-edge collapse operation because it does not introduce new vertex positions but rather sub-samples the original mesh.[18] Thus, it enables the construction of nested hierarchies on unstructured meshes that can facilitate further applications.[19]

The central ingredient of our framework is the global downsampling that removes an independent set of vertices (i.e. vertices which are not connected by an edge) per level of detail. This technique was presented by Kobbelt[7] in order to achieve optimal performance with his multi-level smoothing algorithm. Our global downsampling method selects an independent set of vertices, labled *odd vertices*, to be removed by a sequence of half-edge collapses. The inverse operation (i.e. global upsampling) required for the recontruction operation re-inserts the previously removed vertices to create a mesh topologically identical to the initial one. The original PM framework always reconstructs the original mesh. In order to generate a smooth approximation of the original mesh, we split each step of the PM refinement into a topological operation (vertex insertion) and a geometric operation which places the re-inserted vertices at their original position. In our subdivision scheme, the geometric operation is performed by the relaxation operator presented in section 2 to create a smooth approximation of the fine mesh.

Different methods can be used to select the odd vertices, and thus the half-edge collapses to perform. Our algorithm performs an incremental selection by selecting one odd vertex and locking all adjacent vertices, labled *even vertices*. The selection ends when no more vertices can be selected. By selecting an odd vertex in order to remove it, we also select the even vertex it will merge with. In other words, we directly select an half-edge collapse. The selection of the odd vertices can be done with different methods. We choose the Quadric Error Metric from Garland et al.[20] because it minimizes the length of the details and retains the visual appearance of the simplified mesh.

3.2. Decomposition

The decomposition creates a sequence of levels of resolution, where each of them is composed by an approximation mesh and a set of details associated to the vertices. The creation of one level of resolution requires two major steps: simplification and subdivision. The simplification creates the coarse mesh according to the initial mesh. The subdivision approximates the fine mesh according to the coarse mesh. The details are given as the difference between the initial and the approximated fine mesh. The creation of a level of resolution for a mesh can be broken down into four steps:

- **Select odd vertices.** Starting from a fine mesh $\mathcal{M}^k$, a set vertices, *odd vertices*, is selected to be removed. The remaining vertices composing the coarse mesh are the *even vertices*. The selection of the odd vertices is done using a global downsampling method discribed before.

- **Relax vertices**. The vertices of the mesh are relaxed to create a smooth approximation of the fine mesh. All the attributes a_n of every vertices are relaxed:

$$\forall v_i^k \in \mathcal{M}^k, \quad Rf_n(v_i^k) = \sum_{j \in \mathcal{V}_1(i)} \lambda_{i,j}^k . f_n(v_j^k) \quad \text{with} \quad n \in [1, N], \tag{4}$$

  where the $\lambda_{i,j}$ are defined in Equation 2, and are stored for the reconstruction. The attributes of every vertex v_i^k are relaxed according the surface geometry represented by the $\lambda_{i,j}$ coefficients.

- **Compute details.** We compute the details needed for the reconstruction. The details are given by the difference between the relaxed and the original vertices:

$$\forall v_i^k \in \mathcal{M}^k, \quad d_n^{k+1}(v_i^k) = f_n(v_i^k) - Rf_n(v_i^k) \quad \text{with} \quad n \in [1, N], \tag{5}$$

  where $d_n^{k+1}(v_i^k)$ represents the detail of the attribute a_n of the vertex v_i^k. The details are computed for the N attributes of every vertex v_i^k. We note $d^{k+1}(v_i^k)$ the set $\{d_n^{k+1}(v_i^k)\}$ of details of the attributes of the vertex v_i^k, with $n \in [1, N]$. And we note $\mathcal{D}^{k+1}$ the set $\{d^{k+1}(v_i^k)\}$ of details for every vertex.

- **Remove odd vertices.** Finally, the odd vertices are removed by a sequence of half-edge collapses to create a coarse approximation $\mathcal{M}^k$ of the initial mesh $\mathcal{M}^k$.

Note that the details are computed for the odd and the even vertices. This leads to an overrepresentation of the fine mesh (*i.e.* an optimal representation would compute the details only for the removed vertices). Starting from a fine mesh $\mathcal{M}^k$, the coarse level of resolution is given by the couple $(\mathcal{M}^{k+1}, \mathcal{D}^{k+1})$.

3.3. Reconstruction

The reconstruction is simply the reverse operation of the decomposition. A fine mesh is created using a coarse mesh and the details attached to it. Attributes of one vertex v_i^k of the fine mesh $\mathcal{M}^k$ are reconstructed by relaxing the attributes using the $\lambda_{i,j}$ coefficients computed during the decomposition, and by adding the details of the coarse level. The reconstruction of a fine level of resolution can be broken down into three steps:

- **Insert odd vertices.** First the odd vertices are re-inserted using vertex splits into the coarse mesh $\mathcal{M}^{k+1}$ so that the new mesh $\widetilde{\mathcal{M}}^k$ has exactly the same topology as the initial fine mesh $\mathcal{M}^k$.
- **Reconstruct odd vertices.** The newly re-inserted vertices are reconstructed using the following reconstuction formula for the odd vertices:

$$f_n(\widetilde{v_i}^k) = \sum_{j \in \mathcal{V}_1(i)} \lambda_{i,j}^k . f_n(v_j^{k+1}) + d_n^{k+1}(v_i^k) \quad \text{with} \quad n \in [1, N]. \tag{6}$$

  The reconstruction of the odd vertices creates an intermediate mesh $\widetilde{\mathcal{M}}^k$ where the even vertices are the vertices of the coarser mesh $\mathcal{M}^{k+1}$ and the odd vertices have been recontructed according to the even vertices.

- **Reconstruct even vertices.** Finally the even vertices are reconstructed from the vertices $\widetilde{v_j}^k$ of the intermediate mesh $\widetilde{\mathcal{M}}^k$ using the following reconstruction formula:

$$f_n(v_i^k) = \sum_{j \in \mathcal{V}_1(i)} \lambda_{i,j}^k . f_n(\widetilde{v_j}^k) + d_n^{k+1}(v_i^k) \quad \text{with} \quad n \in [1, N]. \tag{7}$$

  The even vertices of the fine mesh $\mathcal{M}^k$ are recontructed according to the vertices of the approximate fine mesh $\widetilde{\mathcal{M}}^k$.

4. APPLICATIONS

In this section, we present different applications of the multiresolution analysis of irregular meshes. Geometric applications are presented, such as filtering and denoising. Applications involving attribute management are shown, such as adaptive simplification.

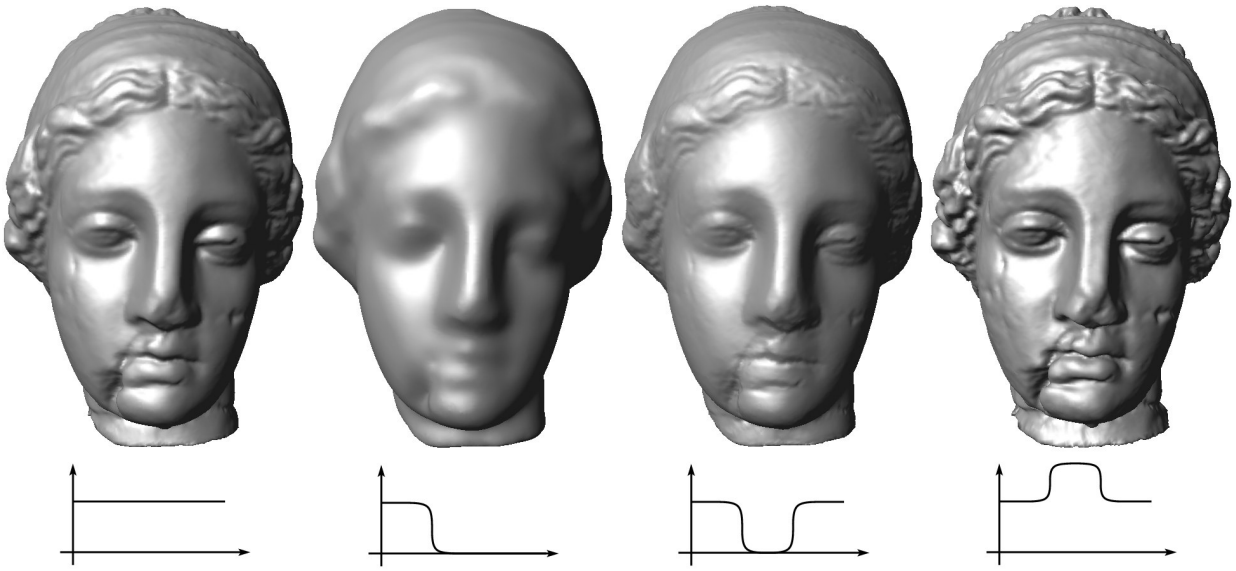
4.1. Filtering

We can apply some signal processing filters to a multiresolution model. Figure 5 shows three different filters applied on the Venus model[†] (Fig. 5(a)). Low pass filtering (Fig. 5(b)) is performed by setting to zero all the details of the 6 finest levels. We see that the low pass filter achieves a smooth result, as expected. Stopband filtering (Fig. 5(c)) is performed by setting to zero all the details from level 6 to level 9. We see that intermediate details are removed, while fine details are preserved. Enhancement filtering (Fig. 5(d)) is performed by multiplying by two all the details from level 6 to level 9. We see that mid-level details are emphazised. These results are similar to those presented in Daubechies' work.[21]

4.2. Denoising by Soft Thresholding

In this section we extend the concept of soft-thresholding to 3D meshes. Wavelet shrinkage is a popular method used to denoise data. The idea is to transform the data into the wavelet basis, where the large coefficients are mainly the signal, and smaller ones represent the noise. By suitably modifying these coefficients, the noise can be removed from the data. It is important to understand the wavelet shrinkage methods as a denoising algorithm and not as a smoothing algorithm. Denoising attempts to remove whatever noise is present and to retain whatever signal is present regardless of the frequency content, while smoothing removes high frequencies and retains low frequencies.[22] The soft thresholding method[23,24] has proved its efficiency and its robustness for denoising appplications. Given a threshold λ for monodimensional data d, the rule $\delta_\lambda(d) = sgn(d)\max(0, |d| - \lambda)$ defines soft thresholding, the operator δ_λ nulls all values of d for which $|d| \leq \lambda$ and shrinks toward the origin by an amount of λ all values for which $|d| > \lambda$.

[†]http://www.cyberware.com/samples



(a) Initial model. (b) Low pass filter. (c) Stopband filter. (d) Enhancement filter.

Figure 5. Filtering of the Venus head.

In the setting of multiresolution analysis of irregular meshes, geometric details $\mathbf{d}$ are represented by a vector in 3D. Therefore, the soft thresholding rule has to be adapted to 3D vectors, but the principle remains the same. The operator nulls all values of $\mathbf{d}$ for which $\|\mathbf{d}\| \leq \lambda$ and shrinks the vector $\mathbf{d}$ toward the origin by an amount of λ if $\|\mathbf{d}\| > \lambda$. Given threshold λ for data $\mathbf{d}$ in 3D space, the soft thresholding is defined by:

$$\delta_\lambda(\mathbf{d}) = \begin{cases} \mathbf{0}, & \|\mathbf{d}\| \leq \lambda \\ \mathbf{d} - \frac{\mathbf{d}}{\|\mathbf{d}\|}.\lambda, & \|\mathbf{d}\| > \lambda \end{cases}. \tag{8}$$

Figure 6 shows results of denoising the Jared model 6(a). This model was acquired with a RIEGL LMS-Z210 time-of-flight laser range scanner mounted on a van. The noise comes from the vibrations of the van and the limited accuracy of the scanner. This model was denoised using the soft thresholding on the five finest levels of details (Fig. 6(b)). Notice that the model is nicely smoothed, while keeping important geometrical features such as sharp edges. Soft thresholding is a fast, efficient denoising method and gives good results. Moreover, our methods requires only one pass over the model.

4.3. Adaptive simplication

Our decomposition method creates a mesh pyramid, and so a hierarchy of meshes at different levels of resolution. Bonneau[6] discussed the advantages of threshold reconstruction against level reconstruction. Level reconstruction results from applying the reconstruction algorithm (section 3.3) up to a certain level. Threshold reconstruction inserts vertices in the mesh according to the length of their details. Only vertices with a detail length superior to a given threshold will be inserted into the reconstructed mesh. Moreover our method used the PM framework which has proved its efficiency for view-dependent and adaptive visualization of irregular meshes.[25] An adaptative reconstruction (and thus simplification) of a multiresolution mesh can be done as follows:

- Sort the list of removed vertices according to the length of their details for each type of detail.
- Reconstruct a new mesh by inserting vertices with detail length superior to a given threshold.

To avoid the construction of a degenerate mesh, for each vertex inserted we have to also insert the vertices to which the newly inserted vertex depends on in the coarser levels. The reconstruction can be guided by the user in case of the presence of several attributes. The user can specify only one type of attribute to be taken in account for the reconstruction (geometrical attributes should always be considered).

Figure 6(c) shows the simplified version of the model in Fig. 6(b), where the color attributes have been considered for the reconstruction. The threshold can be fixed manually or it can be fixed interactively in case of visualization. The threshold changed according to given criteria (such as framerate, visibility).

5. CONCLUSION AND FUTURE WORK

We have presented a new multiresolution analysis for irregular meshes with multiple attributes. We have presented a fast, local surface relaxation operator, and shown how to extend this operator to the attributes. A complete multiresolution analysis framework has been detailed. We introduced denoising through soft thresholding for irregular meshes. This method requires only one pass over the model, and keeps important features such as sharp edges. Results show the efficiency of our framework through a number of applications including filtering, denoising, and adaptive simplification. Future works will address feature detection and adaptive visualization of complex models.

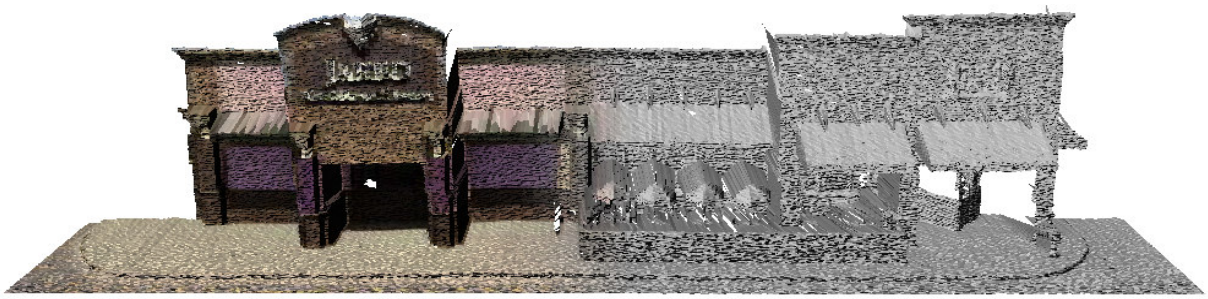
ACKNOWLEDGMENTS

This work was supported by the University Research Program in Robotics under grant DOE-DE-FG02-86NE37968, by the DOD/TACOM/NAC/ARC Program, R01-1344-18, and by FAA/NSSA Program, R01-1344-48/49. The authors would like to thank Yohan Fougerolle for his helpful comments, and Brad Grinstead for the Jared model.

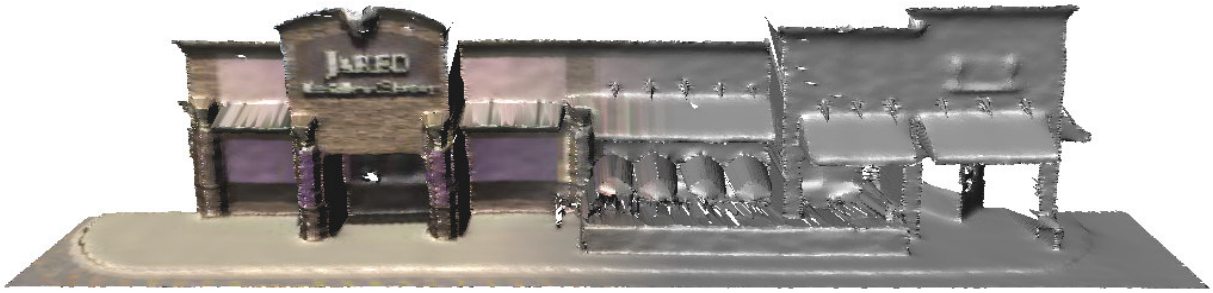
REFERENCES

1. J. D. Walter and C. G. Healey, "Attribute preserving dataset simplification," in *Proceedings of IEEE Visualization*, pp. 113–120, 2001.
2. S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, second ed., 1999.
3. E. J. Stollnitz, T. D. DeRose, and D. H. Salesin, *Wavelets for Computer Graphics: Theory and Applications*, Morgan Kaufmann, 1996.
4. M. Lounsbery, T. DeRose, and J. Warren, "Multiresolution analysis for surfaces of arbitrary topological type," *ACM Transactions on Graphics* **16**(1), pp. 34–73, 1997.
5. D. Zorin, P. Schröder, and W. Sweldens, "Interactive multiresolution mesh editing," in *Proceedings of ACM SIGGRAPH*, pp. 259–269, 1997.
6. G.-P. Bonneau, "Multiresolution analysis on irregular surface meshes," *IEEE Transactions on Visualization and Computer Graphics* **4**(4), pp. 365–378, 1998.
7. L. Kobbelt, S. Campagna, J. Vorsatz, and H.-P. Seidel, "Interactive multi-resolution modeling on arbitrary meshes," in *Proceedings of ACM SIGGRAPH*, pp. 105–114, 1998.
8. H. Hoppe, "Progressive meshes," in *Proceedings of ACM SIGGRAPH*, pp. 99–108, 1996.
9. I. Guskov, W. Sweldens, and P. Schröder, "Multiresolution signal processing for meshes," in *Proceedings of ACM SIGGRAPH*, pp. 325–334, 1999.
10. G. Taubin, "A signal processing approach to fair surface design," in *Proceedings of ACM SIGGRAPH*, pp. 351–358, 1995.
11. L. Kobbelt, "Discrete fairing," in *Proceedings of IMA Conference on the Mathematics of Surfaces*, pp. 101–131, 1997.
12. M. Desbrun, M. Meyer, P. Schröder, and A. Barr, "Implicit fairing of irregular meshes using diffusion and curvature flow," in *Proceedings of ACM SIGGRAPH*, pp. 317–324, 1999.
13. I. Guskov, "Multivariate subdivision schemes and divided differences," tech. rep., Department of Mathematics, Princeton University, 1998.
14. M. Meyer, M. Desbrun, P. Schröder, and A. Barr, "Discrete differential-geometry operators for triangulated 2-manifolds," in *Proceedings of Visualization and Mathematics*, 2002.

15. R. Schneider and L. Kobbelt, "Geometric fairing of irregular meshes for freeform surface design," *Computer Aided Geometric Design* **18**(4), pp. 359–379, 2001.
16. A. Hubeli and M. Gross, "Multiresolution methods for non-manifold models," *IEEE Transactions on Visualization and Computer Graphics* **7**(3), pp. 207–221, 2001.
17. P. Burt and E. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications* **31**(4), pp. 532–540, 1983.
18. L. Kobbelt, S. Campagna, and H.-P. Seidel, "A general framework for mesh decimation," in *Proceedings of Graphics Interface*, pp. 43–50, 1998.
19. J. Wu and L. Kobbelt, "Fast mesh decimation by multiple-choice techniques," in *Proceedings of 7th International Fall Workshop on Vision, Modeling, and Visualization*, pp. 241–248, 2002.
20. M. Garland and P. Heckbert, "Surface simplification using quadric error metrics," in *Proceedings of ACM SIGGRAPH*, pp. 209–216, 1997.
21. I. Daubechies, I. Guskov, P. Schröder, and W. Sweldens, "Wavelets on irregular point sets," *Philosophical Transactions of the Royal Society* **357**(1760), pp. 2397–2413, 1999.
22. C. Taswell, "The what, how, and why of wavelet shrinkage denoising," *IEEE Computing in Science & Engineering* **2**(3), pp. 12–19, 2000.
23. D. Donoho, "De-noising by soft-thresholding," *IEEE Transactions on Information Theory* **41**(3), pp. 613–627, 1995.
24. I. Fodor and C. Kamath, "Denoising through wavelet shrinkage: an empirical study," *SPIE Journal of Electronic Imaging* **12**(1), pp. 151–160, 2003.
25. H. Hoppe, "View-dependent refinement of progressive meshes," in *Proceedings of ACM SIGGRAPH*, pp. 189–198, 1997.



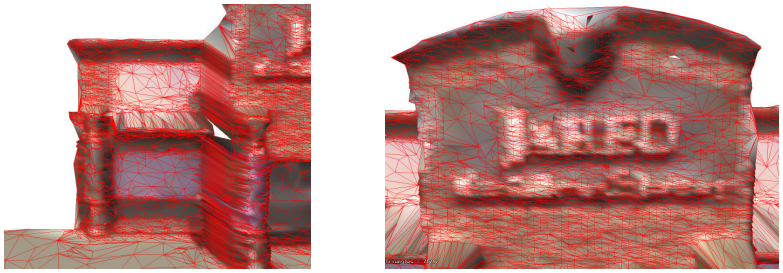
(a) Initial noisy model with colors (184,429 faces).



(b) Denoising using soft thresholding (184,429 faces).



(c) Adaptive simplification of the denoised model (71,717 faces).



(d) Magnification of the model in (c).

Figure 6. Denoising and simplification of the Jared model.