The displaced subdivision surface reconstruction from unorganized points

Won-Ki Jeong*  Young-Sik Jeong**  Chang-Hun Kim*

*Dept. of Computer Science & Engineering, Korea University
{wkjeong, chkim}@cgvr.korea.ac.kr
**Dept. of Computer Engineering, Wonkwang University
ysjeong@wonkwang.ac.kr

Abstract

In this paper we propose a new mesh reconstruction scheme that produces a displaced subdivision surface directly from unorganized points. The displaced subdivision surface is a new mesh representation that defines a detailed mesh with a displacement map over a smooth domain surface, but original displaced subdivision surface algorithm needs an explicit polygonal mesh since it is not a mesh reconstruction algorithm but a mesh conversion (remeshing) algorithm. The main idea of our approach is that we sample surface detail from unorganized points without any topological information. For this, we predict a virtual triangular face from unorganized points for each sampling ray from a parameteric domain surface. Direct displaced subdivision surface reconstruction from unorganized points has much importance since the output of this algorithm has several important properties: It has compact mesh representation since most vertices can be represented by only a scalar value. Underlying structure of it is piecewise regular so it can be easily transformed into a multiresolution mesh. Smoothness after mesh deformation is automatically preserved. We avoid time-consuming global energy optimization by employing the input data dependant mesh smoothing, so we can get a good quality displaced subdivision surface quickly.

1. Introduction

By the improvement of optical and mechanical technology, to acquire accurate surface information from a real object has become commonplace. Such technologies includes laser scanner, mechanical probe, and structured light give output in the form of an unorganized point-cloud. To be adapted for our graphics system, unorganized points should be converted into a smooth surface or a polygonal mesh. Hence, there is lots of literature on mesh reconstruction algorithm.

Mesh reconstruction algorithm gives a dense, seamless irregular polygonal mesh as an output. Such meshes are appropriate for expressing fine surface detail, but notorious for their huge amount of data. So, many optimization algorithms – simplification, multiresolution, compression, etc. – have been developed.

Displaced subdivision surface, proposed by Aaron lee [13], is the new mesh representation that expresses a detailed model as a scalar displacement over a smooth subdivision surface. This representation dramatically reduces the amount of data since it requires only a scalar value for expressing a 3D vertex. This can be thought as a lossy-compression since displaced subdivision surface is an approximation of original mesh, not exact the same mesh. Besides this, parameterization and smoothness of domain surface is automatically defined by a stationary subdivision scheme, and this representation can be converted easily into a bump map to improve rendering performance. Even though this representation has lots of benefits, it needs a reconstructed polygonal mesh to be sampled surface detail from. In the original displaced subdivision surface conversion pipeline, we should have 3D polygonal mesh before the sampling detail process. Normally, a 3D polygonal mesh can be acquired through two processes – the 3D scanning process and the mesh reconstruction process.

Figure 1: Comparison of conversion pipeline. White box: object. Grey box: operation. Thick arrow: ordinary conversion. Dotted arrow: our method.

In this paper, we suggest a new mesh reconstruction algorithm that accepts an unorganized range of points as an input and returns a displaced subdivision surface as a result. The main idea is to sample fine surface detail from unorganized points without
any topological information. Original displaced subdivision surface algorithm samples surface detail from an explicit polygonal mesh, but our algorithm does it from unorganized points. This has much importance in two ways. First, we can skip the mesh reconstruction process in the displaced subdivision surface conversion pipeline (Figure 1). Second, the output of this algorithm has several important properties that an ordinary mesh doesn’t have: It has compact mesh representation since most vertices can be represented by only a scalar value. Underlying structure of it is piecewise regular so it can be easily transformed into a multiresolution mesh. Smoothness after mesh deformation is automatically preserved. The biggest challenge is to estimate the unknown surface of given input points for sampling detail.

Our algorithm works relatively fast since we employ an intuitive geometric method – predicting a virtual triangular face from unorganized points for each sampling ray – to find a correct displacement and avoid the global optimization. To enhance surface quality, we define the input data dependant Laplacian operator. With this, we can smooth a result mesh with minimizing overall shape changes. A result mesh has a lot of applications – wavelet based multiresolution analysis, mesh compression, editing, rendering, and animation.

This paper consists of as follows. First, we briefly review previous work including reconstruction, subdivision, and displacement map, then introduce displaced subdivision surface algorithm. Next, we introduce our main algorithm, followed by experimental result. Finally, we conclude our paper with future work.

2. Previous Work

2.1 Mesh reconstruction

There’s a large literature on 3D reconstruction from unorganized points in the computer vision and graphics fields. Most reconstruction schemes have focused on the approximation of a smooth parametric surface to given points or the derivation of a zero-set of an implicit function [1, 16]. Recently, the extracting triangular mesh from a given set of points is studied. Hoppe [6, 7, 8] proposed the arbitrary 3D mesh reconstruction from unorganized points. He introduced volume-based reconstruction with optimization of energy functions. He employed modified Loop subdivision scheme to optimize result. This method has several advantages - the ability to reconstruct arbitrary topological mesh, optimal and robust result – but it needs much computation. Suzuki [9] proposed a subdivision surface fitting algorithm that uses the limit surface property of approximating subdivision scheme. He changes the shape of control mesh at every level of subdivision to maximally fit limit surface into input points. This method needs small computation, but the result mesh lacks surface detail. Ramamoorthi [18] proposed the reconstruction of a generative model from unorganized points. This method gives a robust result, but there’s topological limitation and lack of detail.

2.2 Subdivision surface

Subdivision is the process of making smooth and detailed mesh by iteratively dividing of a given coarse control mesh. It is divided into several categories by the type of mesh and division method as follows [21].

2.2.1 Approximating scheme

The approximating subdivision scheme produces the result mesh that does not interpolate the initial control mesh. The result mesh only approximates the control mesh. Each vertex on the initial control mesh acts like a control point of a spline surface. A subdivision process consists of two steps: the splitting step and the averaging step. The splitting step is the connectivity-updating step by introducing new vertex for each edge. The average step is repositioning all vertices by the weight sum of their 1-neighbors.

Loop scheme [15] is the most popular subdivision scheme for triangular mesh. It is based on three-way box spline, and the result mesh is \( C^2 \) continuous except for extraordinary points. Catmull & Clark scheme [2] is the subdivision scheme for rectangular mesh based on tensor product bicubic spline. Continuity property is similar to Loop scheme.

Figure 2: Example of Loop subdivision

(a) Control mesh (F:60 V:32)  (b) Level 5(F:61440 V:30722)

2.2.2 Interpolating scheme

The interpolating subdivision scheme produces the result mesh that interpolates all the initial control vertices. So, once a vertex position is determined, it is the limit position. This scheme does not induce shrinkage, so it is useful when we need the result that interpolating user specified positions. But this scheme lacks of continuity property. Butterfly scheme proposed by Dyn, Gregory, and Levin [3] and Denis Zorin [20] are the interpolating subdivision scheme for triangular meshes. This scheme needs wide subdivision mask compare to Loop scheme, and only \( C^1 \) continuity is guaranteed for ordinary vertices. The limit surface produced by this scheme is not a piecewise-polynomial surface. Kobbelt [10] proposed an interpolating subdivision scheme for quadrilateral mesh. This scheme is extended to adaptive approach.

2.3 Displacement map

Recently, several algorithms that convert an arbitrary mesh into fitted smooth surfaces and displacements are proposed. Main benefit of this work is that the displacement map can be easily transformed into bump map to enhance rendering process. Krishnamurthi and Levoy [12] proposed a method of smooth surface fitting to a polygonal mesh. They manually divide the input mesh into several sections, and fit B-spline surfaces to them. After fitting process, they sample fine surface detail with displacement vectors. This method can give good smooth surface fitting, but displacements are three dimensional vectors.
Moreover, this method needs a lot of manual process for dividing the input mesh. Recently, Aaron lee [13] proposed another surface fitting algorithm that uses subdivision scheme to produce a smooth parameteric surface and a scalar displacement value for sampling surface detail. Since this algorithm constructs the main structure of our paper, we’ll review this algorithm in detail in the following section.

3. Displaced Subdivision Surface

The displaced subdivision surface, proposed by Aaron lee [13], is a new mesh representation that represents an arbitrary mesh as a scalar-valued displacement over a smooth subdivision surface.

![Mesh](Image)

**Figure 3:** Sampling displacement from parameteric surface

The main idea is to making smooth domain surface and the displacement function by existing stationary subdivision scheme. The displaced subdivision surface conversion process consists of three steps as follows.

1. Using QEM [5], we simplify an input mesh into an initial control mesh and keep the parameterization by MAPS [14]. For sampling surface detail, we restrict an edge collapse that doesn’t preserve vertex normal direction.

2. When we get the initial control mesh, the optimize control mesh in order to the domain surface fits well to the original mesh by Hoppe’s methods [7].

3. We subdivide the optimized control mesh into the smooth domain surface. Then we sample the displacement map by computing the signed distance from the limit surface of the smooth domain surface to the original surface along the limit surface normal.

The displaced subdivision surface is an extremely compact representation for mesh. Since a domain surface is defined by subdivision process, only a coarse control mesh should be stored. A displacement is sampled along the limit surface normal direction by a scalar value, so we can reduce the storage for saving displacements by 1/3 compared with other algorithm.

The underlying structure of this representation is also a benefit. Since the displaced subdivision surface has the piecewise-regular structure, it is easily transformed into the multiresolution mesh by the wavelet-based analysis [4, 17]. A scalar displacement can be the proper wavelet coefficient in this representation.

Since the smoothness of the domain surface is guaranteed by the subdivision scheme, this representation is robust on editing and deformation for animation.

4. Reconstruction algorithm

Our algorithm consists of four steps as follows.

1. First, we make an initial control mesh from input points. We introduce a manually defined control mesh for arbitrary topology as well as totally automatic schemes for planar or spherical topology.

2. We subdivide the initial control mesh defined above by Loop subdivision scheme and make a parameteric domain surface fitting to control mesh.

3. We sample fine surface detail on parameteric domain surface from input points by shooting rays from the domain surface along the limit surface normals. For each surface limit normal, we make a virtual plane that intersects it and sample displacement.

4. To enhance the quality of the reconstructed mesh, we define the new Laplacian operator that penalizes moving away from input points and smooth the result mesh with it.

4.1 Making initial control mesh

Making control mesh is the critical process in our algorithm since a control mesh defines the shape of parameteric domain surface. First thing that we should do is to recognize topological structure of the unknown arbitrary surface of given points. Previous work [6] employed volumetric approach to solve this problem. This is the reasonable way to extract topological information, but the volume-based iso-surface extraction is not an easy task. It generates too much data, and the surface quality is not good. In this paper, we restrict input topology to planar or spherical surface. We left the method for arbitrary topology for the future work.

For the planar model, we get two points – the biggest and the smallest x, y, and z values – and make the rectangle whose dihedral edge consists of these two points. Figure 4 shows the example of making a control mesh of planar model.

![Mesh](Image)

**Figure 4:** Making planar control mesh of terrain data

For the spherical model, we make a bounding octahedron for input points. We divide the octahedron until we get the desired resolution, and update its vertex position by that of an input point sampled by the ray shooting from a center of the points. Figure 5 shows the example of the process of making a spherical control mesh.
Since the fully automatic method doesn’t consider local complexity or curvature, it may not capture local surface detail well. To compare with our automatic method, we manually defined a control mesh that has more faces in the complex area.

### 4.2 Making parameteric domain surface

This step is to make a smooth parameteric domain surface. Since original displaced subdivision surface has employed Loop subdivision scheme [15] to generate a smooth surface, we follow the same subdivision scheme. To successfully capture fine surface detail, a domain surface and input points should be close enough. Since Loop subdivision induces shrinkage, we import subdivision surface fitting scheme [9] to modify control mesh so that the domain surface fits well to input points.

By [15], we find a limit position \( p_i^- \) of each vertex \( p_i \) on the current domain surface \( M \). Then we find the closest input point \( q_i \) of each limit position \( p_i^- \). The resultant force \( r_i \) is defined as follows:

\[
r_i = (q_i - p_i) - R \sum_{j \in i} (p_j - p_i)
\]

where

\[
R_i = (\frac{3}{8\beta} + k)^{-1},
\]

\[
\beta = \begin{cases} 
\frac{3}{16} & k = 3 \\
\frac{1}{k} \left( \frac{5}{8} - \frac{3}{4} \cos \left( \frac{2\pi}{k} \right) \right) & k \geq 3 \\
\text{val}(p_i) & k = 1
\end{cases}
\]

We can modify each vertex \( p_i \) of the control mesh by the following iterative approximation method

\[
p_i' = p_i + \lambda r_i
\]

Optimal \( \lambda \) value is 0.8 by experiments. After this fitting process, we subdivide the control mesh until we get the smooth domain surface that has similar complexity of input points by comparing the number of total vertices.

### 4.3 Sampling surface detail

In the original displaced subdivision surface sampling, they sought to compute the signed distance from the limit point of each vertex on the domain surface to the original surface along the normal. Since the sampling target is polygonal mesh, it is easy to find an intersection point between the normal and the original surface. But in our algorithm, we don’t have any face or connectivity information. Hence, we make a virtual face for each normal to sample surface detail. The idea is simple. We find three points close to each normal. Then we test whether the normal vector intersects the triangle made with these three points or not. If it does, we calculate the signed distance from the domain surface to this triangle along the normal. If not, we add the next near vertex and make combinations to generate other triangular faces until we get an intersected face. If we get multiple intersected faces, we choose the one that has the smallest area. Figure 7 shows this sampling process of one face.

Since there’s no connectivity information of the input points, we have a problem in efficient data management. In [13], author used a spatial hierarchy for finding an intersection position of the normal and the original surface. To efficient searching process, we employed the neighbor information queue for every input point where we store up to \( n \) close points of a given input point. \( n \) is given by user, depending on the complexity of input points. If input points are dense, we take big \( n \) value. If not, we take small \( n \) value. In our experiments, we take \( n \) value of a given point as the number of points that the distance is less than 0.1.

### 4.4 Enhancing surface quality

The reconstructed mesh in the previous section is an approximation of input points. This can be the output of our algorithm as itself. But if we need a more optimized result, we can enhance
the surface quality by the Laplacian smoothing. Ordinary mesh smoothing is the Gaussian filtering defined as follows [19]:
\[ M^{n+1} = M^n + \lambda L(M^n) \]
\[ L(x_i) = \frac{1}{m} \sum_{j \in N_i(x_i)} (x_j - x_i) \] (3)

For a given mesh \( M^n \), we can smooth the mesh while preserves the overall shape. It is done by the iterative summation of the current mesh and a Laplacian of every vertex. The discrete Laplacian of a vertex on a polygonal mesh is a weight sum of its 1-neighbor vertices [11]. Since this Laplacian induces the shrinkage, we define the new Laplacian operator, the input points dependent Laplacian.

\[ L(x_i) = (x_{\text{nearest}} - x_i) + \frac{1}{m} \sum_{j \in N_i(x_i)} (x_j - x_i) \] (4)

\( x_{\text{nearest}} \): the nearest to \( x_i \) among the input points

\( N_i(x_i) \): 1-neighbor vertices of \( x_i \)

The vector \( x_{\text{nearest}} - x_i \) means the outer force from \( x_i \) to the input points. The vector sum \( \frac{1}{m} \sum_{j \in N_i(x_i)} (x_j - x_i) \) is the vector from \( x_i \) to the 1-neighbor plane, which means the local curvature.

This Laplacian is defined by the sum of two forces, one minimizes the local curvature and the other minimizes the moving away from the input points. By this Laplacian, we can get the maximally smoothed surface without the distortion of overall shape.

5. Result

We’ve implemented our algorithm on Pentium II 350 with 128M-ram environment. OpenGL, Visual C++, and MFC library were used. The results are shown in the figure 9-10. Control meshes of spock and cat model were made manually. Following table shows the size of data and execution time.

<table>
<thead>
<tr>
<th>Model</th>
<th>Data</th>
<th>Execution time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Input # V</td>
<td>Domain # V</td>
</tr>
<tr>
<td>Geometric</td>
<td>9218</td>
<td>4098</td>
</tr>
<tr>
<td>Spock</td>
<td>16386</td>
<td>8002</td>
</tr>
<tr>
<td>Cat</td>
<td>23362</td>
<td>8002</td>
</tr>
</tbody>
</table>

For multiresolution modeling, an arbitrary mesh must be converted into a regular structured mesh. This conversion process is called remeshing [4]. Most existing remeshing schemes use 1-4 subdivision to make a regular structure. Since our algorithm is a sampling process from a smooth subdivision surface to input points, underlying surface structure is piecewise regular – this means the result mesh has subdivision connectivity. So, we can apply our result mesh to multiresolution analysis without remeshing. Figure 8 shows a displaced subdivision mesh that is produced by our method. You will see that our result shows the piecewise regular structure. In multiresolution analysis, wavelet coefficients are extracted at the analysis process. Loundsbery et al [17] suggested the vector-based detail coefficient extraction. Since our algorithm gives scalar value displacement for surface detail, it can be applied to wavelet coefficient directly.

Figure 8. Underlying piecewise-regular structure of cat model reconstructed by our method. Left: irregular structure. Right: piecewise-regular structure.

6. Conclusion and Future Work

In this paper, we introduced a new mesh reconstruction algorithm that extracts a displaced subdivision mesh directly from unorganized points. Original displaced subdivision surface conversion is the sampling from a polygonal mesh, so it needs the mesh reconstruction process for unorganized points. We can reconstruct a displaced subdivision surface simple and fast by adopting sampling technique from unorganized points and the input point dependent Laplacian to avoid the global optimization. The result mesh is the compact representation and ready to be applied into the wavelet based multiresolution analysis without remeshing since it samples scalar-value displacement and has subdivision connectivity. The underlying structure and smoothness of the domain surface are simply defined by the stationary subdivision scheme. The result mesh has lots of application – mesh editing, animation, rendering, etc.

For future work, we are working on the extension of this algorithm for the reconstruction of arbitrary topological model. More robust sampling fine surface detail technique is also being studied.

7. References


Figure 9: Result of the reconstruction process of cat model range data.

(a) Input range data  (b) Domain surface  (c) Surface detail sampling  (d) Final result

Figure 10: Result of the reconstruction of spock model range data.

(a) Domain surface with input range data  (b) Surface detail sampling near nose area  (c) Final result