From Scattered Samples to Smooth Surfaces

Kai Hormann[1]

California Institute of Technology

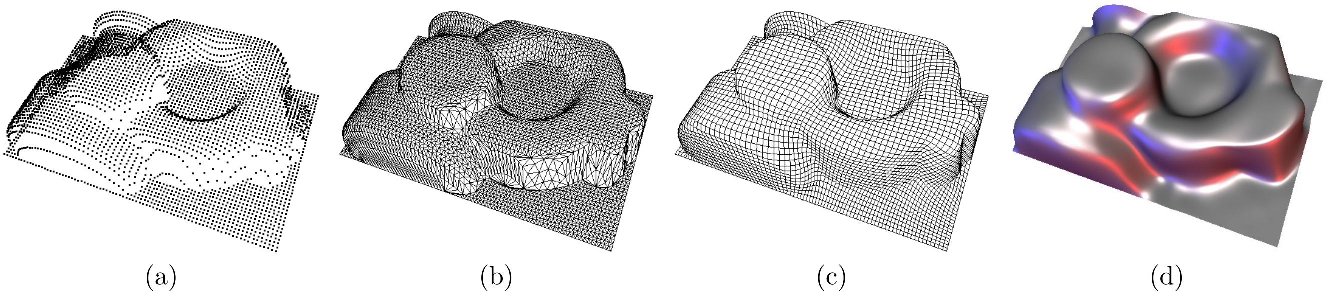

Figure 1: A point cloud with 4,100 scattered samples (a), its triangulation with 7,938 triangles (b), remesh with 80×48 quadrilaterals (c), and smooth approximation with a 23×15 bicubic tensor product B-spline surface (d).

ABSTRACT

The problem of reconstructing smooth surfaces from discrete scattered sample points arises in many fields of science and engineering and the data sources include measured values (laser range scanning, meteorology, geology) as well as experimental results (engineering, physics, chemistry) and computational values (evaluation of functions, finite element solutions, numerical simulations). We describe a processing pipeline that can be understood as gradually adding order to a given unstructured point cloud until it is completely organized in terms of a smooth surface. The individual steps of this pipeline are triangulation, remeshing, and surface fitting and have in common that they are much simpler to perform in two dimensions than in three. We therefore propose to use a parameterization in each of these steps so as to decrease the dimensionality of the problem and to reduce the computational complexity.

CR Descriptors: I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling; J.6 [**Computer-Aided Engineering**]: Computer-Aided Design (CAD); G.1.2 [**Approximation**]: Approximation of surfaces and contours

Keywords: Surface Reconstruction, Parameterization, Triangle Mesh, Remeshing, Spline Surface

[1]Caltech, MS 256-80, 1200 E. California Blvd., Pasadena CA 91125, USA; phone: 001-626-395-2907, fax: 001-626-792-4257, e-mail: hormann@cs.caltech.edu

1 INTRODUCTION

The problem we consider can be stated as follows: given a set $V = \{v_i\}_{i=1,\ldots,n}$ of points $v_i \in \mathbb{R}^3$, find a surface $S : \Omega \to \mathbb{R}^3$ that approximates or interpolates V. If the points are totally unstructured and not associated with any additional information, then the first step usually is to triangulate the points, in other words, to find a piecewise linear function S that interpolates V. This introduces a first level of organization as it defines the topological structure of the point cloud. For example, the eight vertices of a cube could be triangulated by four triangles forming two parallel squares or by twelve triangles forming a closed surface (see Figure 2). In Section 3 we discuss how this problem can be solved by constructing a parameterization of V and then using a standard triangulation method in two dimensions. Note that certain acquisition methods (e.g. laser range scanning) provide connectivity information that allows to use simpler triangulation methods.

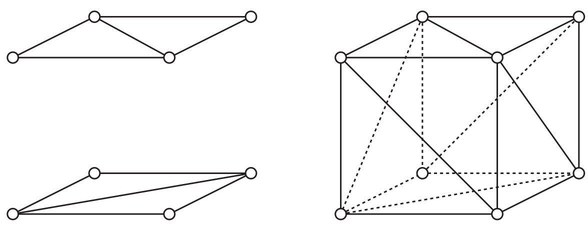

Figure 2: The triangulation of a point cloud is topologically ambiguous.

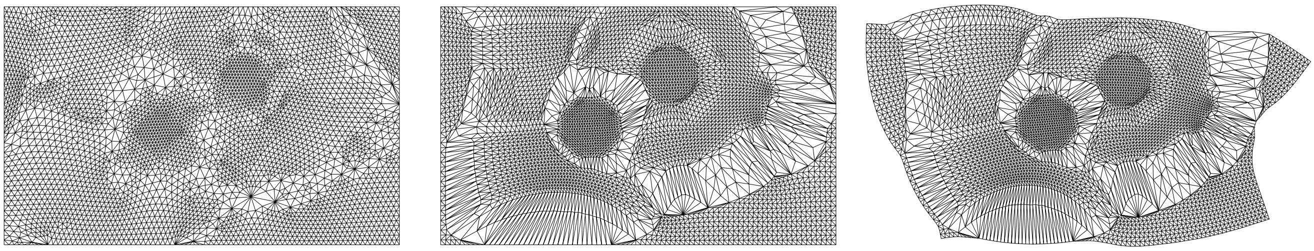

Figure 3: Uniform, shape preserving, and most isometric parameterization of the triangulation in Figure 1 (b).

The process of remeshing can be considered a second level of organization as it approximates a given unstructured triangulation with another triangulation that has a certain regularity, namely subdivision connectivity. Again, the remeshing process is relatively simple to perform in two dimensions and in Section 4 we show how it can be done once a parameterization of the triangulation has been determined. We also mention remeshing with regular quadrilateral meshes as this leads to an efficient indirect smooth surface approximation scheme.

Approximating the given scattered samples with a smooth surface can be viewed as the third level of organization as the surfaces we consider are twice differentiable and so to say regular everywhere. We review the classical variational approach to smooth surface fitting in Section 5 and compare it to the previously mentioned indirect method. A parameterization of the data points is also needed in this third step to set up the smooth approximation problem and we therefore start by explaining how to compute such parameterizations.

2 PARAMETERIZATION

Parameterizing a point cloud V is the task of finding a set of *parameter points* $\psi(v_i) \in \Omega$ in the *parameter domain* $\Omega \subset \mathbb{R}^2$, one for each point $v_i \in V$. A parameterization of a triangulation $\mathcal{T}$ with vertices $V = V(\mathcal{T})$ is further said to be *valid* if the *parameter triangles* $t_i = [\psi(v_j), \psi(v_k), \psi(v_l)]$ which correspond to the triangles $T_i = [v_j, v_k, v_l]$ of $\mathcal{T}$ form a valid triangulation of the parameter points in the parameter domain, i.e. the intersection of any two triangles t_i and t_j is either empty, a common vertex, or a common edge.

A lot of work has been published on parameterizing triangulations over the last years and the most efficient methods can be expressed in the following common framework. Their main ingredient is to specify for each interior vertex v of the triangulation $\mathcal{T}$ a set of weights λ_{vw}, one for each vertex $w \in N_v$ in the neighbourhood of v, where N_v consists of all vertices that are connected to v by an edge. The parameter points $\psi(v)$ are then found by solving the linear system

$$\psi(v) \sum_{w \in N_v} \lambda_{vw} = \sum_{w \in N_v} \lambda_{vw} \psi(w). \tag{1}$$

The simplest choice of weights is motivated by a physical model that interprets the edges in the triangulation as springs and solves for the equilibrium of this network of springs in the plane [12]. These *distance weights* are defined as

$$\lambda_{vw} = \lambda_{wv} = \|v - w\|^{-p}$$

and yield *uniform*, *centripetal*, or *chord length* parameterizations for $p = 0$, $p = 1/2$, or $p = 1$. Note that these weights are positive and thus always give valid triangulations as shown by Tutte [23] for uniform and later by Floater [7] for arbitrary positive weights. However, these parameterizations fail to have a basic property. If the given triangulation is flat, we would expect the parameterization to be the identity, but it turns out that this cannot be achieved for any choice of p.

Another choice of weights that gives parameterizations with this reproduction property are *harmonic* weights [3, 20],

$$\lambda_{vw} = \lambda_{wv} = \cot \alpha_v + \cot \alpha_w,$$

where α_v and α_w are the angles opposite the edge connecting v and w in the adjacent triangles (see Figure 4). But these weights can be negative and there exist triangulations for which the harmonic parameterization is not valid.

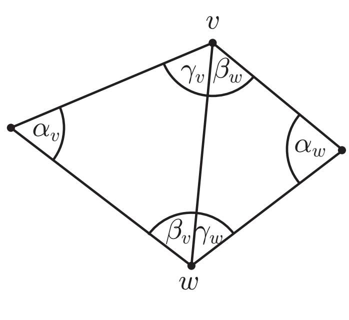

Figure 4: Notation for defining various weights.

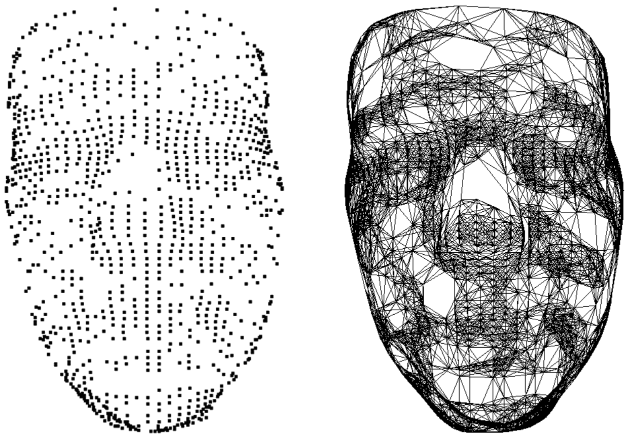

Figure 5: A point cloud with 1,042 samples and the connectivity graph for $k = 16$.

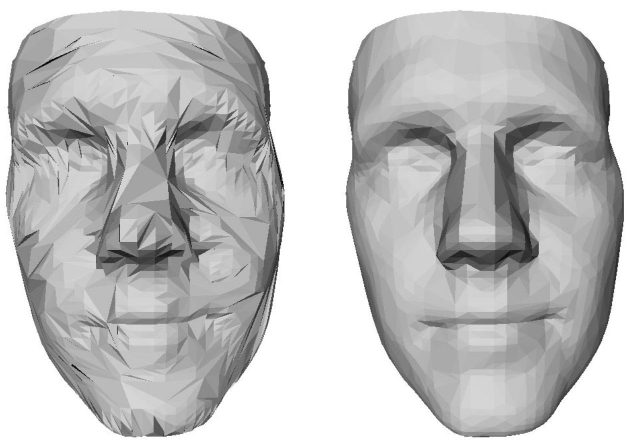

Figure 6: Reconstructed triangulation before and after optimization.

The *shape preserving* weights [4] were the first known to result in parameterizations that meet both requirements, but also the *mean value* weights [6]

$$\lambda_{vw} = (\tan(\gamma_v/2) + \tan(\beta_w/2))/\|v - w\|$$

do. In addition, they depend smoothly on the v_i.

The drawback of these linear methods is that they require at least some of the boundary vertices to be fixed and parameterized in advance and it is not always clear how this is done best. Non-linear methods that overcome this limitation at the expense of higher computation complexity include *most isometric* parameterizations [15] as well as an approach that minimizes the overall angle deformation [22]. Examples of parameterizations are shown in Figures 3, 7, and 9.

3 TRIANGULATION

Floater and Reimers [8] observed that linear parameterization methods can also be used for parameterizing point clouds as they do not require the points to be organized in a globally consistent triangulation. The only information needed to compute distance weights is a set of neighbours N_v for each interior vertex v, and determining the harmonic, shape preserving, and mean value weights additionally requires N_v to be *ordered*.

A simple choice of N_v is the ball-neighbourhood

$$N_v^r = \{w \in V : 0 < \|v - w\| < r\}$$

for some radius r. But especially for irregularly distributed samples, taking the k nearest neighbours as N_v has proven to give better results. Typical values of k are between 8 and 20.

Both choices of N_v can be used to define an ordered neighbourhood by projecting all neighbours into the least squares fitting plane of $v \cup N_v$ and considering the *Delaunay triangulation* of the projected points [8, 9].

Once these neighbourhoods are specified, it is possible to apply one of the linear parameterization methods to determine parameter points $\psi(v_i)$ and use one of the standard methods for triangulating points in two dimensions, e.g. the Delaunay triangulation, to find a triangulation $\mathcal{S}$ of the $\psi(v_i)$. Finally, a triangulation $\mathcal{T}$ of V is obtained by collecting all triangles $[v_j, v_k, v_l]$ for which $[\psi(v_j), \psi(v_k), \psi(v_l)]$ is a triangle in $\mathcal{S}$.

Figures 5–7 show an example where a point cloud was parameterized using 16 nearest neighbours and chord length parameterization. The resulting triangulation was optimized by using an edge flipping algorithm that minimizes mean curvature [2].

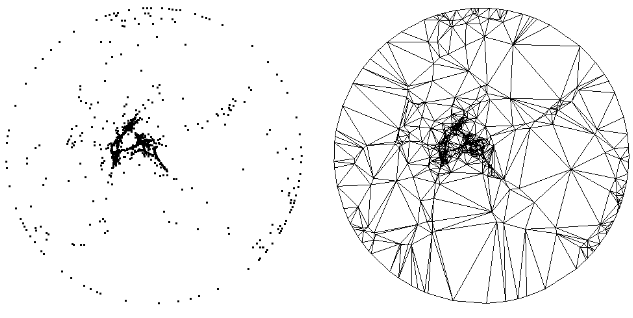

Figure 7: Chord length parameterization of the point cloud in Figure 5 and Delaunay triangulation.

4 REMESHING

Point cloud parameterizations usually have very low quality because the connectivity graph that is used for generating them does not reflect the properties of the surface from which the samples were taken well. Therefore the reconstructed surface in Figure 6 looks crinkly. But after optimization the techniques from Section 2 generate high quality parameterizations which can be further used for other reconstruction methods.

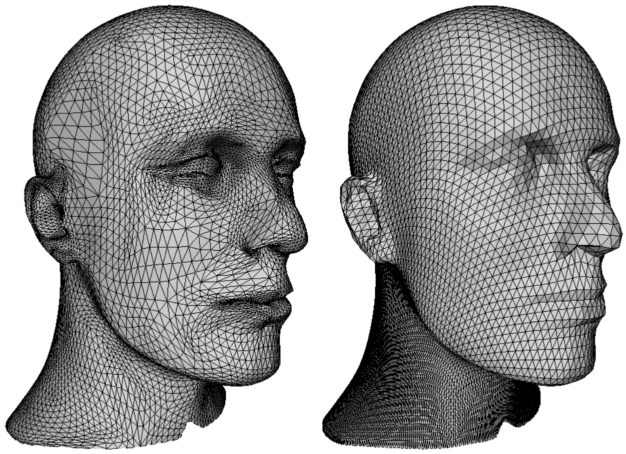

Figure 8: Triangulation with 21,680 triangles and regular remesh with 24,576 triangles.

One important aspect of surface reconstruction is to approximate a given triangulation $\mathcal{T}$ with a new triangulation $\mathcal{T}'$ that has regular connectivity. This process is commonly known as *remeshing* and motivated by the fact that the special structure of the new triangulation allows to apply very efficient algorithms for displaying, storing, transmitting, and editing [1, 18, 19, 21, 25]. The special structure required by these algorithms is *subdivision connectivity*, which is generated by iteratively refining a triangulation dyadically.

With a parameterization at hand, remeshing can easily be performed in the two dimensional parameter space Ω. The simplest approach is to choose Ω to be a triangle and iteratively splitting this triangle into four by inserting the edge midpoints. The vertices of this *planar* remesh are then lifted to $\mathbb{R}^3$ to give the *spatial* remesh $\mathcal{T}'$ by using the parameterization in the following way. First, the *barycentric coordinates* with respect to the surrounding parameter triangle $[\psi(v_j), \psi(v_k), \psi(v_l)]$ are computed for each vertex w of the planar remesh. Then the corresponding vertices v_j, v_k, v_l of $\mathcal{T}$ are linearly interpolated using these coordinates to give the vertex w' of the spatial remesh.

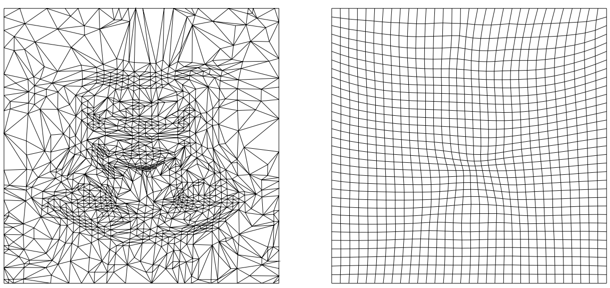

Figure 9: Shape preserving parameterization and quadrilateral remesh of the triangulation in Figure 6.

The quality of the spatial remesh can be improved by smoothing the planar remesh in the parameter domain [17]. For example, the remesh in Figure 8 was generated by iteratively applying a weighted Laplacian to the vertices of the planar remesh where the weights depended on the areas of the triangles in the spatial remesh so as to give a remesh with uniformly sized triangles in the end. This remeshing strategy is not limited to triangulations with subdivision connectivity and can also be used to approximate triangulations with regular quadrilateral meshes [16] (see Figure 9).

5 SURFACE FITTING

Parameterizations are also important for smooth surface reconstruction. Given a function space S that is spanned by k basis functions $B_j : \Omega \to \mathbb{R}^3$, the task is to find the coefficients c_j of an element $F = \sum_j c_j B_j$ of S such that $F(\psi(v_i)) \approx v_i$ for all $i = 1, \ldots, n$.

The quadrilateral remeshing approach allows for a very efficient solution of this problem, namely interpolation at the vertices w' of $\mathcal{T}'$. This indirectly approximates the initially given vertices and requires to solve only a few tridiagonal linear systems if cubic tensor product B-splines are used as basis functions [16].

Figure 10 compares the result of the indirect method to that of the classical variational approach [5, 14, 24]. The idea of the latter is to minimize a weighted combination $\mathcal{E}(F) + \mu \mathcal{J}(F)$ of the ℓ_2 approximation error

$$\mathcal{E}(F) = \sum_{i=1}^{n} \|F(\psi(v_i)) - v_i\|^2$$

and a quadratic smoothing functional $\mathcal{J} : S \to \mathbb{R}$. The surfaces in Figures 10 and 11 were obtained by using the *simplified thin plate energy* [10, 11, 13]

$$\mathcal{J}(F) = \int_\Omega F_{uu}^2 + 2F_{uv}^2 + F_{vv}^2 \, du \, dv.$$

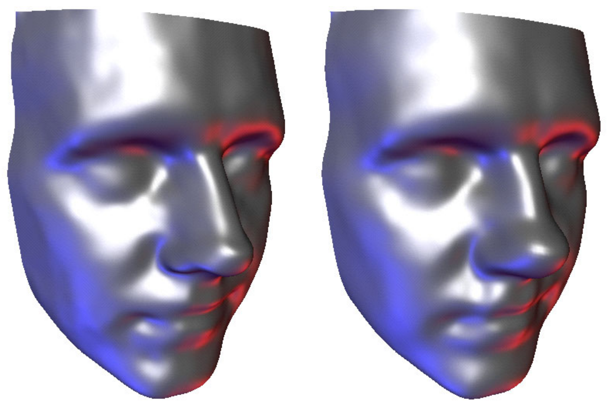

Figure 10: Indirect and smooth least squares approximation of the point set in Figure 5.

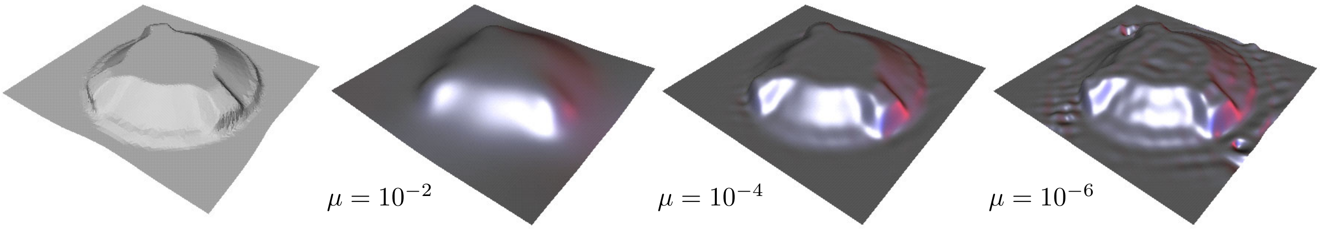

Figure 11: Triangulation with 3,374 vertices and surface reconstruction with different smoothing factors.

REFERENCES

[1] A. Certain, J. Popović, T. DeRose, T. Duchamp, D. Salesin, and W. Stuetzle. Interactive multiresolution surface viewing. In *ACM Computer Graphics (SIGGRAPH '96 Proceedings)*, pages 91–98, 1996.

[2] N. Dyn, K. Hormann, S.-J. Kim, and D. Levin. Optimizing 3D triangulations using discrete curvature analysis. In T. Lyche and L. L. Schumaker, editors, *Mathematical Methods for Curves and Surfaces: Oslo 2000*, pages 135–146. Vanderbilt University Press, 2001.

[3] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. In *ACM Computer Graphics (SIGGRAPH '95 Proceedings)*, pages 173–182, 1995.

[4] M. S. Floater. Parameterization and smooth approximation of surface triangulations. *Computer Aided Geometric Design*, 14:231–250, 1997.

[5] M. S. Floater. How to approximate scattered data by least squares. Technical Report STF42 A98013, SINTEF, Oslo, 1998.

[6] M. S. Floater. Mean value coordinates. *Computer Aided Geometric Design*, to appear.

[7] M. S. Floater. One-to-one piecewise linear mappings over triangulations. *Mathematics of Computation*, to appear.

[8] M. S. Floater and M. Reimers. Meshless parameterization and surface reconstruction. *Computer Aided Geometric Design*, 18:77–92, 2001.

[9] M. Gopi, S. Krishnan, and C. T. Silva. Surface reconstruction based on lower dimensional localized Delauney triangulation. In *Computer Graphics Forum (Eurographics '00 Proceedings)*, volume 19, pages 119–130, 2000.

[10] G. Greiner. Surface construction based on variational principles. In P. J. Laurent, A. Le Méhauté, and L. L. Schumaker, editors, *Wavelets, Images, and Surface Fitting*, pages 277–286. AK Peters, Wellesley, 1994.

[11] G. Greiner. Variational design and fairing of spline surfaces. In *Computer Graphics Forum (Eurographics '94 Proceedings)*, volume 13, pages 143–154, 1994.

[12] G. Greiner and K. Hormann. Interpolating and approximating scattered 3D data with hierarchical tensor product B-splines. In A. Méhauté, C. Rabut, and L. L. Schumaker, editors, *Surface Fitting and Multiresolution Methods*, pages 163–172. Vanderbilt University Press, 1997.

[13] M. Halstead, M. Kass, and T. DeRose. Efficient, fair interpolation using Catmull-Clark surfaces. In *ACM Computer Graphics (SIGGRAPH '93 Proceedings)*, pages 35–44, 1993.

[14] K. Hormann. Fitting free form surfaces. In B. Girod, G. Greiner, and H. Niemann, editors, *Principles of 3D Image Analysis and Synthesis*, pages 192–202. Kluwer Academic Publishers, Boston, 2000.

[15] K. Hormann and G. Greiner. MIPS: an efficient global parametrization method. In P.-J. Laurent, P. Sablonnière, and L. L. Schumaker, editors, *Curve and Surface Design: Saint-Malo 1999*, pages 153–162. Vanderbilt University Press, 2000.

[16] K. Hormann and G. Greiner. Quadrilateral remeshing. In B. Girod, G. Greiner, H. Niemann, and H.-P. Seidel, editors, *Vision, Modeling and Visualization 2000*, pages 153–162. infix, 2000.

[17] K. Hormann, U. Labsik, and G. Greiner. Remeshing triangulated surfaces with optimal parametrizations. *Computer-Aided Design*, 33(11):779–788, 2001.

[18] A. Khodakovsky, W. Sweldens, and P. Schröder. Progressive geometry compression. In *ACM Computer Graphics (SIGGRAPH '00 Proceedings)*, pages 271–278, 2000.

[19] M. Lounsbery, T. DeRose, and J. Warren. Multiresolution analysis for surfaces of arbitrary topological type. *ACM Transactions on Graphics*, 16:34–73, 1997.

[20] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, 2(1):15–36, 1993.

[21] P. Schröder and W. Sweldens. Spherical wavelets: Efficiently representing functions on the sphere. In *ACM Computer Graphics (SIGGRAPH '95 Proceedings)*, pages 161–172, 1995.

[22] A. Sheffer and E. de Sturler. Parameterization of faceted surfaces for meshing using angle based flattening. *Engineering with Computers*, 17:326–337, 2001.

[23] W. T. Tutte. How to draw a graph. *Proc. London Math. Soc.*, 13:743–768, 1963.

[24] M. von Golitschek and L. L. Schumaker. Data fitting by penalized least squares. In J. C. Mason, editor, *Algorithms for approximation II*, pages 210–227, Shrivenham, 1988.

[25] D. Zorin, P. Schröder, and W. Sweldens. Interactive multiresolution mesh editing. In *ACM Computer Graphics (SIGGRAPH '97 Proceedings)*, pages 259–268, 1997.