International Journal of Pure and Applied Mathematics
Volume 55 No. 1 2009, 61-72

SURFACE FLATTENING VIA CONFORMAL MAPPING

Qingbin Li[1], Junxiao Xue[2 §]

[1]Department of Mathematics and Physics
Zhengzhou Institute of Aeronautical Industry Management
Zhengzhou, 450015, P.R. CHINA
e-mail: liqingbin82@126.com

[2]Software Technology School
Zhengzhou University
Zhengzhou, 450002, P.R. CHINA
e-mail: xuejx7@yahoo.com

Abstract: In this paper, we present a surface flattening method based on conformal mapping. The method is an iterative procedure, which incrementally flattens a 3-D surface to 2-D domain by region growing. The algorithm is greedy and begins with a seed triangle which can be chosen freely. In each operation, we develop a piecewise linear mapping function that is conformal. In addition, at first step, we initialize the planar triangle which corresponds to the seed triangle. The initialization ensures that the seed triangle flattening process produce no distortion. And in other steps, we reduce the distortion by unifying the vertices in the plane. Our flattening method has low distortion. Experiments show that our method is efficient and fast enough.

AMS Subject Classification: 26A33
Key Words: surface flattening, conformal mapping, parameterization

1. Introduction

Surface flattening is the problem of mapping a 3-D surface into 2-D plane. It plays a prominent role in engineering and manufacturing applications, such as aircraft design, vehicle design, garment design [10], [21], etc. In addition, the

Received: June 19, 2009

§Correspondence author

surfaces used in computer graphics are very often represented as triangular meshes with irregular connectivity and non-uniform sizes, and many computer graphic operations, such as brain imaging [2], [20], 3-D painting, texture mapping [3], [12], require a non-distorted mapping between the 3-D surface points and the 2-D image sample, which is equivalent to a non-warped flattening of the surface.

Ideally, the flattening process between the triangulated surface and the planar triangulation should be an isometric, preserving angles and distances. Unfortunately, except developable surfaces, general open surfaces cannot reach the ideal aim. It has been shown by Gauss in 1828 that an isometric mapping between two surfaces with different intrinsic curvature is not possible [4]. Thus, it is impossible to flatten undevelopable surface to the plane without distortions, because they have different Gaussian curvature.

In this paper, we address the flattening procedure using conformal mapping, which is a 2-D parameterization of a 3-D surface such that angles are preserving. Conformal mapping is bijective. Moreover, the elements of the first fundamental form (E,F,F) are transformed as $(\rho E,\ \rho F,\ \rho G)$ for some positive function depending on the point of the surface. So the intrinsic metric on the original surface is transformed. For this reason, shape is locally preserved, and both the distances and the areas are only changed by a scaling factor. We propose a simple and recursive algorithm in this paper. The surface patch we discuss is presented as triangular meshes, and it is topologically equivalent to a disk.

Our approach is guided by three principles: (1) For each triangle, we convert its global coordinates (x_i,y_i,z_i) representation to its local coordinate system. The system is defined with the triangle's normal as the z-axis. So the triangle is embedded in the local x-y plane. (2) We consider the mapping of the 2-D x-y local coordinates to the corresponding triangle in the u-v plane, which is a linear system of equations. (3) Starting the flattening procedure from a seed triangle. In each step, a new vertex which neighbors with the already flattened patch is chosen for flattening. The choosing of seed triangle is freedom, and in the beginning of the flattening procedure, we initialize the corresponding triangle in the plane to make sure that the length of the triangle's edges is equal to the 3-D triangle's edges. The selection of new vertex for flattening should follow the criteria that the flattening boundary length may be as small as possible. The criteria prevents the creation of thin long patches (strips).

The proposed method has several advantages. First, it is locally shape preserving and near equilateral. So the mapping in our approach is approximately isometric (conformal and equiareal). Experiments show that the distortion value achieved is small. Second, our approach does not provide the optimal solution. It is greedy and in each step the solution just depending on a linear system. Therefore, our method is efficient and can be performed quickly.

The rest of the paper is organized as follows. In Section 2 we review some related work. Section 3 gives a brief sketch of conformal mapping theory. Our algorithm is described in Section 4. Section 5 concludes the paper and suggests some topics for future work.

2. Related Works

The flattening of a triangular 3-D mesh, which provides a bijective mapping between the mesh and a triangulation of a planar polygon, plays an important role in parameterization and texture mapping. Floater et al [5], provide an extensive survey of the state of the art in mesh parameterization research. In [6], Floater investigate a graph theory based parameterization for tessellated surfaces for the purpose of smooth surface fitting; his parameterization (actually a planar triangulation) is the solution of linear systems based on convex combination. In [9], Hormann and Greiner use Floater's algorithm as a starting point for a highly non-linear local optimization algorithm which computes the positions for both interior and boundary nodes based on local shape preservation criteria. They called their approach: Most Isometric Parametrizations (MIPS). The method is promising, but it is not clear if the procedure is guaranteed to converge to a valid solution.

Wolfson et al [22] and Schwartz et al [17] introduce a flattening method based geodesic distance. They first use a computationally intensive way for finding the geodesic distance between pairs of points on the surface. Then, they use a specific MDS (see [11], Multi-Dimensional Scaling) approach to flatten the surface using these geodesic distances, and by minimizing the functional presented by Sammon in[16]. The work by Gil Zigelman et al [24] analytically finds an embedding of an open mesh in the plane by a MDS (Multi-Dimensional Scaling) method that optimally preserves the geodesic distances between mesh vertices. However, finding minimal geodesic distances between points on a continuous surface is a classical and difficult problem in differential geometry. Whole the method above involves high computational complexity.

The angle based flattening (ABF) method presented by Sheffer et al [18] is based on the observation that the set of angles of a 2-D triangulation uniquely defines the triangulation up to global scaling and rigid transformations. They defined an angle preservation metric directly in terms of angles. It first computers the parameterization in angle space and only then converts it into 2-D coordinates. In addition to avoiding flips, its important advantage is that in addition to closely preserving the angles it typically produces parameterizations with low area (and stretch) deformation. However, the optimization procedure used by ABF is numerically expensive. Recently, some researchers discussed methods to speed up ABF but do not provide an implementation. Zayer et al [23] proposed a different strategy for solving the non-linear optimization problem by ABF. In [19], Sheffer et al stated a highly efficient of the ABF method using a hierarchical technique.

Some surface flattening methods based on the strain-energy minimization [3], [13]. They assume the original 3-D surface has zero energy, i.e. without wrinkles or stretches, while the 2-D pattern is sought that minimizes the deformation energy. Actually, their non-linear optimization scheme handles in a few seconds about thousand vertices. By the limitation of the irregular mesh utilized in the above algorithms, the anisotropic material is hardly simulated. C.C.L. Wang et al [21] utilize a woven-like regular quadrilateral mesh model, which greatly facilitates the simulation of anisotropic material based surface flattening.

Conformal maps preserve both the magnitude and sense of angles between arbitrary arcs [14]. Thus, angle preservation is typically addressed from the conformal point of view. B. Lévy et al [12] give a quasi-conformal parameterization method, based on a least squares approximation of the Cauchy-Riemann equations. The objective function minimizes angle deformation. In addition, N. Ray and B. Lévy [15] introduce HLSCM (Hierarchical Least squares Conformal Map), an efficient parameterization method for large meshes. The work by Gu [7] and Steven Haker et al [8] have created flattened representations or visualizations of the cerebral cortex or cerebellum. These works indicate that if a quasi-length and area-preserving mapping is desired, the conformal mapping technique is a very reasonable starting point, since it preserves local geometry.

We note that, beyond the above work, what makes our approach practical and easy to implement is the key observation that the conformal transformation is based on a system of linear equations and the flattening procedure is presented as an iterated extension.

3. Brief Sketch of Conformal Mapping Theory

A real function of a real variable can easily be visualized by means of a graph. If $y=f(x)$, we interpret x and y, respectively, as the abscissa and the ordinate of a point in a rectangular coordinate system. However, in the case of complex-valued functions of a complex variable, some changes are clearly called for. Since the values of both the function and the variable are complex numbers, a point representation of the function $w=f(z)$ would require four numbers, thus necessitating the use of four-dimensional space. Since such a space is not accessible to our geometric visualization, but little would be gained by a geometric representation of this type. A geometric representation of the function $w = f(z)$ is obtained by regarding z and w as points in two different planes – the z-plane and the w-plane – and by interpreting the function $w=f(z)$ as a mapping of points in the z-plane onto points in the w-plane. If $f(z)$ is regular and single-valued in a domain D, we shall say that the set of points in the w-plane which corresponds to the points of D by means of the mapping $w=f(z)$ is the map of D as given by this function; we shall also use the word image in this connection [14].

The name conformal mapping for the mapping associated with a regular analytic function is derived from a property by which it is characterized. A mapping is conformal if it preserves the angle between two differentiable arcs. To show that the mapping affected by a regular analytic function is indeed conformal, we proceed as follows. Suppose $f(z)$ is regular in the neighborhood of a point $z=z_0$ at which $f'(z_0) \neq 0$. The point $z = z_0$ is the terminal of two differentiable arcs α and β, the angle between their tangent vectors $\alpha'(z)$ and $\beta'(z)$ at point z_0 is the same as the angle between the tangent vectors $(f \circ \alpha)'(z_0)$ and $(f \circ \beta)'(z_0)$, i.e. the image of α and β under f. This means that f is conformal at point z_0. If f is conformal for all points in D it is called a conformal mapping, which is equivalent to the Cauchy-Riemann equation [1] such that:

$$\frac{\partial f}{\partial \overline{z}} = 0. \tag{1}$$

Considering now $z = x+iy$ and $w = u+iv$, we educe a 2-D function $f(x, y)$ from the function $w = f(z)$, parameterized by (x, y) that returns a 2-D point (u, v). The Cauchy-Riemann equation is that:

$$\frac{\partial f}{\partial x} + i\frac{\partial f}{\partial y} = 0. \tag{2}$$

This implies:

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y} \quad \text{and} \quad \frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}. \tag{3}$$

4. Region Growing Algorithms

We demonstrate our surface flattening algorithm in this section. At the beginning, we give a brief overview and then elaborate on its different components. The concretion of the algorithm is presented in the end.

In this paper, we consider the bounded and simply-connected mesh surface S. Our algorithm is an iterative procedure that incrementally flattens the mesh surface by growing patches around a seed triangle. The seed triangle is randomly selected and embedded on a plane without any distortion. Its three edges define the initial front-patch. In each subsequent iteration, the algorithm examines all the triangles adjacent to the front-patch. Each such triangle has two vertices that have already been mapped onto the plane, and one free vertex. We embed this triangle onto the plane along with the front-patch. When no more vertices can be added to the current patch, the algorithm redefines the front-patch, which is the boundary of the current patch that has been flattened. Then the algorithm starts with the new front-patch.

For our application, since the input is a set of discrete 3-D points on S, in each triangle embedding process, we are looking for a mapping $f : (x, y, z) \rightarrow (u, v)$. The mapping should be approximately isometric, which implies conformal and close to equiareal. The problem can be reduced to $R^2 \rightarrow R^2$ by considering how to map the triangles of a 3-D mesh to their corresponding triangles in the u-v plane in parameter space. First, we embed the triangle in the local x-y plane which is spanned by a local orthonormal basis. Second, we consider a $R^2 \rightarrow R^2$ conformal mapping $f : (x, y, z) \rightarrow (u, v)$, which embeds the triangle onto the u-v plane. The added triangles are also checked for intersections with the planar patch.

Consider now a triangulation $\triangle = \{[1 \dots n], \tau, (p_j)_{1 \leq j \leq n}\}$, where $[1 \dots n]$, $n \geq 3$, corresponds to the vertices, where τ is a set of n' triangles represented by triples of vertices, and where $p_j \in R^3$ denotes the geometric location at the vertex j. We suppose that each triangle is provided with a local orthonormal basis, where (x_1, y_1), (x_2, y_2), (x_3, y_3) are the coordinates of its vertices in this basis. Thus, the triangle is embedded in the local x-y plane. We visualize the embedding in Figure 1.

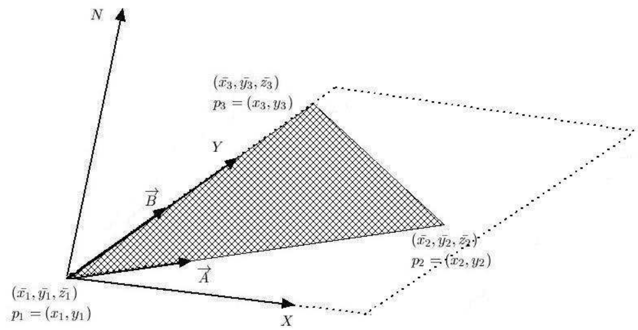

Figure 1: Triangle's local coordinate system

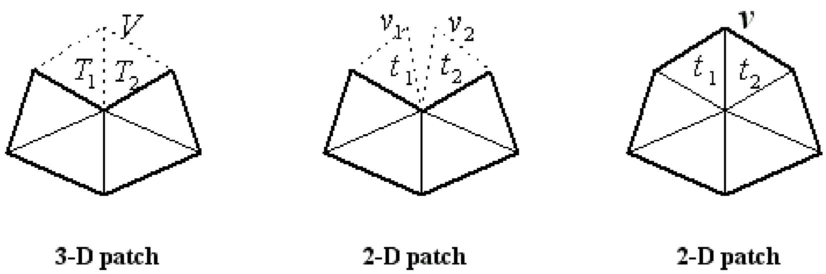

Figure 2: Vertices unifying

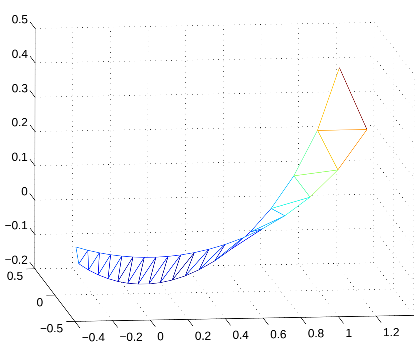

Figure 3: 3-D snail-patch1

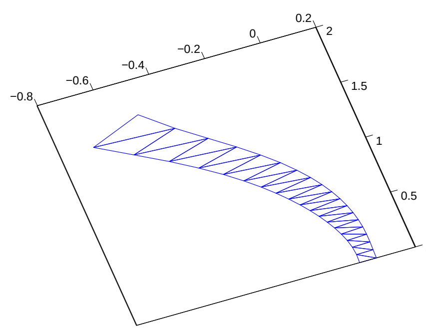

Figure 4: Planar snail-patch1

The triangle's vertices are $p_j(x_j, y_j, z_j)$, $j = 1 \ldots 3$.

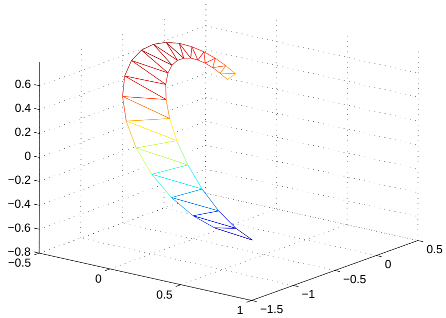

Figure 5: 3-D snail-patch2

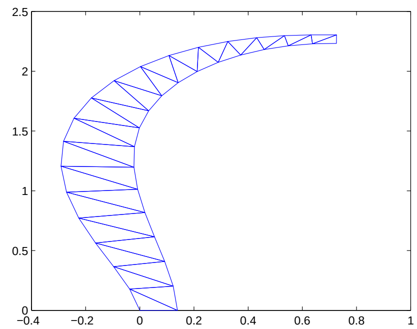

Figure 6: Planar snail-patch2

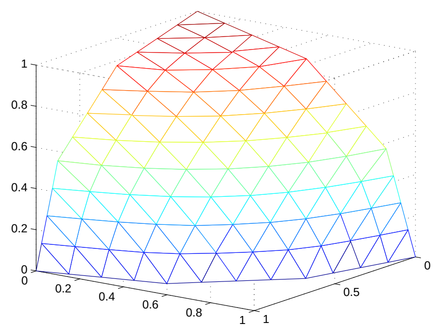

Figure 7: 3-D tri-patch

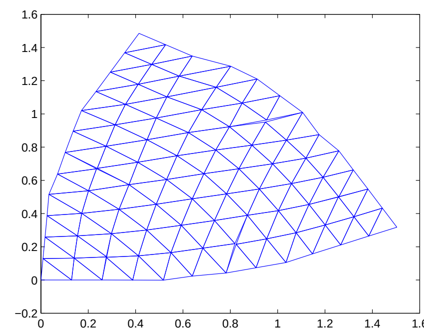

Figure 8: Planar tri-patch

Suppose:

$$\overrightarrow{A} = p_2 - p_1, \quad \overrightarrow{B} = p_3 - p_1, \tag{4}$$

and

$$N = A \times B / \parallel A \times B \parallel, \quad X = N \times B / \parallel N \times B \parallel, \quad Y = N \times X / \parallel N \times X \parallel, \tag{5}$$

X and Y are the triangle's local orthonormal basis. Then:

$$p_1' = (0,0), \quad p_2' = (A \cdot X, A \cdot Y), \quad p_3' = (B \cdot X, B \cdot Y), \tag{6}$$

p_1',p_2' and p_3' are the coordinates of the triangle's vertices in this basis.

A triangle to triangle mapping is defined by a unique affine transformation between the original and destination triangle. If we consider the affine mapping $f(p) \mapsto q$, where $p = (x, y)$ and $q = (u, v)$, where $T_{p_1 p_2 p_3}$ is the source triangle

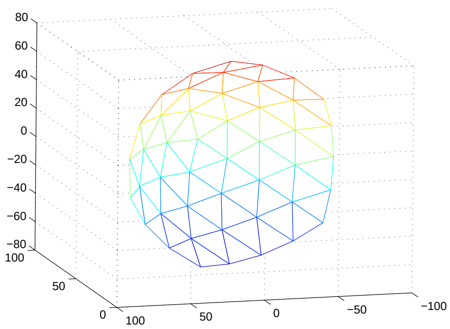

Figure 9: 3-D ball-patch

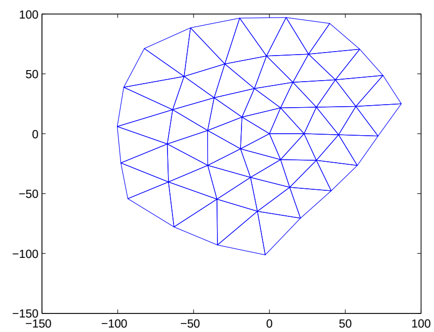

Figure 10: Planar ball-patch

and $T_{q_1q_2q_3}$ is the destination, then the two triangles' vertices are related as:

$$f(p) = \frac{area(T_{pp_2p_3})q_1 + area(T_{p_1pp_3})q_2 + area(T_{p_1p_2p})q_3}{area(T_{p_1p_2p_3})}. \tag{7}$$

The partial derivatives (triangle gradient) of this equation are as follows:

$$\frac{\partial f}{\partial x} = \frac{(y_2 - y_3)q_1 + (y_3 - y_1)q_2 + (y_1 - y_2)q_3}{2area(T_{p_1p_2p_3})}, \tag{8}$$

$$\frac{\partial f}{\partial y} = \frac{(x_3 - x_2)q_1 + (x_1 - x_3)q_2 + (x_2 - x_1)q_3}{2area(T_{p_1p_2p_3})}. \tag{9}$$

This triangle gradient can be used to formulate the Cauchy-Riemann equations stated in equation 3. This can be written as follows:

$$\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} = \frac{1}{2A_T}(\Delta y_1 \cdot u_1 + \Delta y_2 \cdot u_2 + \Delta y_3 \cdot u_3 + \Delta x_1 \cdot v_1 + \Delta x_2 \cdot v_2 + \Delta x_3 \cdot v_3) = 0 \tag{10}$$

$$\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = \frac{1}{2A_T}(\Delta y_1 \cdot v_1 + \Delta y_2 \cdot v_2 + \Delta y_3 \cdot v_3 - \Delta x_1 \cdot u_1 - \Delta x_2 \cdot u_2 - \Delta x_3 \cdot u_3) = 0. \tag{11}$$

Here $\Delta x_1 = x_2 - x_3$, $\Delta x_2 = x_3 - x_1$, $\Delta x_3 = x_1 - x_1$, and Δy_i is defined similarly from equations (8) and (9); A_T is the area of the triangle defined by $T_{p_1p_2p_3}$.

Solving the system of equations (10) and (11) we will find the appropriate (u_i, v_i), which are conformal on $f(x, y) \mapsto (u, v)$. To obtain a unique solution for the system, two vertices of the triangle in u-v plane should be constrained. We will give a detailed state in the follow. For the seed triangle, we fix one vertex of it as the origin of the $u - v$ plane and fix another vertex of it in the u-axis. In addition, the edge which is formed by the two vertices that have been fixed must be equiareal to the corresponding edge in the 3-D mesh surface. Thus, there is no distortion for the seed triangle flattening. For the other triangle, since its two vertices appear in the flattened patch, the constraint is satisfied naturally.

As presented above, in each step of the flattening procedure, we attempt to embed a free vertex V that is adjacent to the front-patch. Let $T_1, T_2, \ldots, T_k$ be the triangles incident to V that share an edge with the front-patch. We would like to map V to a point v in the plane, so as to minimize the maximal distortion caused to the triangles T_i. The optimal position v is found using a local relaxation. We first compute k candidate positions $v_1, v_2, \ldots, v_k$ in the plane, each obtained by dealing with each of the triangles T_i separately. The point v is then obtained as the weighted average of the candidate positions $v_1, v_2, \ldots, v_k$, where the weights are proportional to the areas of the triangles T_i. Note that, this choice of mapping prevents triangle flipping.

5. Conclusion

We propose an incremental flattening procedure, in each step it only needs to solve a system of linear equations. And the mapping in the flattening process is approximately isometric. Thus, our method is fast and efficient for applications whose main requirement is low distortion. However, the surface we considered is simply-connected and especially has low curvature. It is the localization of our technique. For the surface which is multi-connected and has high curvature, it needs segmentation at first, and this may be our future direction of research.

We apply our method to some data sets. Figure 3 – Figure 6 are snail-like patches. Figure 7 and Figure 8 are triangle-like patches. Figure 9 and Figure 10 are ball-like patch. All the patches are simple-connected and do not have high curvatures. In addition, the left patches are 3-D patches, and the right patches are the corresponding 2-D planar patches.

References

[1] L.V. Ahlfros, *Complex Analysis*, Third Edition, McGraw-Hill, Inc., New York (1979).

[2] S. Angenent, S. Haker, A. Tannenbaum, R. Kikinis, Laplace-Beltrami operator and brain flattening, *IEEE Trans. Medical Imaging*, **18**, No. 8 (August 1999), 700-711.

[3] C Bennis, J.M. Vézien, G Iglésias, Piecewose surface flattening for non-distorted texture mapping, *Computer Graphics*, **25**, No. 4 (July 1991), 237-246.

[4] M.P.D. Carmo, *Differential Geometry of Curves and Surfaces*, Prentice Hall, Englewood Cliffs, NJ (1976).

[5] M.S. Floater, K. Hormann, Recent advances in surface parameterization, In: *Multiresolution in Geometric Modeling*, 259-284.

[6] M.S. Floater, Parameterization and smooth approximation of surface triangulations, *Comput Aided Geom. Des.*, **14**, No. 3 (1997), 231-250.

[7] X. Gu, Y. Wang, T.F. Chan, P.M. Thompson, S. Yau, Genus zero surface conformal mapping and its application to brain surface mapping, *IEEE Trans. Medical Imaging*, **23**, No. 8 (August 2004), 949-958.

[8] S. Haker, S. Angenent, A. Tannenbaum, R. Kikinis, G. Sapiro, Conformal surface parameterization for texture mapping, *IEEE Transactions on Visualization and Computer Graphics*, **6**, No. 2 (2000), 181-189.

[9] K. Hormann, G. Greiner, MIPS: An efficient global parameterization method, *Curve and Surface Design*, St. Malo (1999), Vanderbilt University Press (1999), 153-162.

[10] F. Jing, A. Joneja, K. Tang, Modeling wrinkles on smooth surfaces for footwear design, In: *CAD'04 Conference*, Patteya Beach, Thailand, May 24-28 (2004).

[11] J.B. Kruskal, M. Wish, *Multidimensional Scaling*, Sage Publication, Beverly Hills, California (1978).

[12] B. Lévy, S. Petitjean, N. Ray, J. Maillot, Least squares conformal maps for automatic texture atlas generation, *ACM Trans. Graphics*, **21**, No. 3 (2002), 362-371.

[13] J. McCartney, B.K. Hinds, B.L. Seow, D. Gong, An energy based model for the flattening of woven fabrics, *Journal of Materials Processing Technology*, **107** (2000), 312-318.

[14] Z. Nehari, *Conformal Mapping*, Dover Publications, Inc. New York (1952).

[15] N. Ray, B. Lévy, Hierarchical least squares conformal map, In: *PG '03: Proceedings of the 11-th Pacific Conference on Computer* (2003).

[16] J.W. Sammon, A nonlinear mapping for data structuce analysis, *IEEE Transactions on Computers*, **18**, No. 5 (May 1969), 401-409.

[17] E.L. Schwartz, A. Shaw, E. Wolfson, A numerical solution polyhedral surfaces, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **11**, No. 9 (September 1989), 1005-1008.

[18] A. Sheffer, E. de Sturler, Parameterization of faceted surfaces for meshing using angle based flattening, *Engineering with Computers*, **17** (2001), 326-337.

[19] A. Sheffer, B. Lévy, ABF++: Fast and robust angle based flattening temporary version, *In Print*.

[20] G. Stylianou, G. Farin, Crest lines for surface segmentation and flattening, *IEEE Transactions on Vis. and C.G.*, **10**, No. 5 (2004), 536-544.

[21] Char C.L. Wang, K. Tang, M.M.F. Yuen, Freeform surface flattening based on fitting a woven mesh model, *Comput Aided Des.*, **37** (2005), 799-814.

[22] E. Wolfson, E.L. Schwartz, Computing minimal distances on polyhdral surface, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **11**, No. 9 (September 1989), 1001-1005.

[23] R. Zayer, C. Roessl, H.P. Seidel, Variations on angle based flattening, In: *Proceedings of Mingle Workshop* (2004).

[24] G. Zigelman, R. Kimmel, N. Kiryati, Texture mapping using surface flattening via multi-dimensional scaling, *IEEE Trans. Vis. and Comp. Graph.*, **9**, No. 2 (2002), 198-207.