A thesis submitted in partial satisfaction of the
requirements for the degree of
Master of Computer Science and Engineering
in the Graduate School of
the University of Aizu

Shape Modeling with Dynamic Meshes

by

Shin Yoshizawa

March 2001

The thesis titled

Shape Modeling with Dynamic Meshes

by

Shin Yoshizawa

is reviewed and approved by:

Main referee

Associate Professor Date

Alexander G. Belyaev

Professor Date

Daming Wei

Assistant Professor Date

Roman Ďurikovič

The University of Aizu

March 2001

Contents

List of Figures

Acknowledgement

I wish to express my appreciation to Dr. A. G. Belyaev for his excellent advice and diligent efforts to guide me through this project. Also, I would like to thank deeply Dr. A. G. Belyaev for his encouragement, his advice and discussions with him having become the big energy, motivation, and vitality to me in not only this thesis but my graduate school life.

I also express my gratitude for Dr. D. Wei and Dr. R. Ďurikovič who carefully read my thesis and gave many usefull advice.

I would like to thank Mr. Y. Ohtake to provide a few polygonal models with programming advice.

However, all mistakes that remain are my own.

Abstract

The thesis presents a set of new shape modeling methods based on geometric variational concepts. The underlying idea of the methods consist of using shape evolution processes. For computational purposes, smooth shapes are approximated by triangle meshes and discrete evolutions of meshes, dynamic meshes, are considered.

The methods developed in the thesis contribute to the areas of shape smoothing and restoration, geometric feature analysis and extraction, and mesh regularization. A comparison of the methods with previous techniques and algorithms is done. It is demonstrated that the proposed methods have several mathematically-desirable qualities that improve existing mesh fairing techniques.

The list of new shape modeling methods developed in the thesis consists of the following.

Discrete Laplace-Beltrami Operator. A new and robust discrete approximation of the Laplace-Beltrami operator is proposed. Its applications to mesh smoothing and regularization are considered.

Bi-Laplace-Beltrami Flow. The bi-Laplace-Beltrami flow, a new surface flow defined via two Laplace-Beltrami operator iterations, is introduced. Discrete approximations of the flow are proposed. Applications to mesh regularization and fairing are considered.

Discrete Elastica Flow. A new discrete implementation of the elastica flow is proposed. The discrete elastica flow turns out to be very useful for the shape modeling and restoration purposes.

Curvature Visualization on Surface via Pastel and Bipolar Color Mapping. New surface coloring schemes are proposed for visualizing shape curvature features.

Discrete Gaussian Curvatures. Two new and robust discrete approximations of the Gaussian curvature on a triangle mesh are invented. The approximations are based on the Laplace-Beltrami operator and a weighted angle deficit estimation, respectively.

The shape modeling methods considered and developed in the thesis are implemented in the Surface Flow Simulator, a Java3D program.

Keywords: *discrete surface flow, Laplace-Beltrami operator, surface energy functional, nonlinear diffusion, multi-grid approach, surface smoothing, mesh regularization, discrete fairing*

Chapter 1

Introduction

Shape Modeling and Analysis which is research field of arbitrary shape visualization, design, deformation and analysis on the computer is very useful for implementing the technique of Differential, Topological and Computational Geometry on the computer. It is very important to manipulate the shape without the constraint in not only Computer Aided Geometric Design, Computer Graphics, Image Processing and Computer Vision which are related directly but also physical simulation and arbitrary data processing. Because the multi-dimensional data are formed shapes, and the operations of shape affect these mathematical and physical motions or deformations. One of the main topic is curvature, which measures how bend an object, approximation and utilization in the shape modeling and analysis.

Although there are sub fields such as the parametric form based spline modeling, the boolean operation based implicit function modeling, the mesh based shape simplifications and the shape reconstructions via geometrical data optimizations in shape modeling and analys, the interested field is a geometric variational modeling for the smooth deformations with fair surface modeling. The fairing is the shape modeling to interpolate boundary conditions smoothly.

In this thesis, new approaches are described for designing surfaces based on the geometric variational concepts with curvature-driven shape deformation techniques. Moreover, three important applications (smoothing, mesh regularization and fair surface modeling) and its numerical implementations which include good mean and Gaussian curvature approximations are developed. Instead of considering smooth continuous surfaces, the triangulated polygonal surfaces are used in the implementations because triangle meshes are becoming extremely popular in computer science applications for approximation of smooth continuous surfaces with arbitrary shape and topology. Numerical experiments provide good results about the curvature approximations, discrete stable surface smoothing, vertex uniformization and fair surface modeling.

1.1 Previous Works

There are many effective works about the surface flows, curvature approximations, shape denoising and fair surface modeling for the triangulated polygonal surfaces.

The geometric variational concept was appeared about 250 yeas ago [24], but this was becomed popular in geometric modeling since 90's [16, 46] because of their robust calculations and development of numerical analysis for partial differential equations. Simultaneously, exclusive methods to solve variational problems were developed by the implementation of the mathematical methods; the surface evolver was developed in [22], 1992. The elastica flow was applied to closed triangulated polygonal surfaces via the surface evolver in [25], 1992. The level set methods and its applications were considered in [17, 50], 1996. The curvature-driven shape deformation techniques were applied to the nonlinear spline curve modeling [1, 40], 1998.

In [12], the discrete curvatures were estimated by the directional curvature tensor matrix and its eigenvalues in 1995. The methods can calculate not only the principal curvatures but also principal directions. In [11, 13], the discrete fair surface modeling were achieved by a signal processing approach in 1995-1996. The detailed umbrella operator with low-pass filtering was analyzed in these works.

At the end of this century, the high precision approximation methods, high speed modeling methods, high-order derivative of curvature methods and additional methods (mesh enhancement, mesh regularization, adaptive smoothing, etc) became the mainstream of geometric variational modeling. The Laplacian flow was exhibited to the interactive multi-resolution modeling in [26], 1998. The high-order derivative of the curvature flow was represented by the level set methods in [9], 1999. The high precision discrete Laplace-Beltrami operator was developed to the surface smoothing in [31], 1999. The high-order curvature derivative was used for the generating fair meshes in [36], 2000. The mesh enhancement flow was developed by the nonliner diffusion of the normal vector in [49], 2000.

1.2 Approaches

The modeling strategy is constructed as following two differential geometry methods.

Geometric Variational Modeling: The geometric variational modeling is a shape design technique based on the surface energy functionals who are constructed by the surface geometric properties:

$$E(\mathbf{S}) = \iint_{\Omega} F(\mathbf{I}, \mathbf{II}, ...)\, dA,$$

where an energy functional $E(\mathbf{S})$ of a surface $\mathbf{S}$ is organized by the surface area A integral domain Ω of a function $F()$ with the surface geometric property arguments (ex. coefficients of the surface first and second fundamental forms $\mathbf{I}$ and $\mathbf{II}$, see Appendix C).

The objective surface is determined by a solution surface of the stationary partial differential equation (Euler-Lagrange equation) which is equivalent to minimize a desirable surface energy functional. The curvature-driven shape deformation method is applied to solve the Euler-Lagrange equation based on the nonstationary partial differential equation. One of the important properties of a surface evolution is that not only a limit surface but the evolving surfaces have useful effects, because the deformation is dominated by the mathematical surface flow.

Curvature-Driven Shape Deformations: The curvature-driven shape deformation method is a shape variation technique based on the following evolutionary system of a nonlinear partial differential equation:

$$\mathbf{S}_t = F\mathbf{N}, \tag{1.1}$$

where $\mathbf{S}_t$ is a derivative of the surface $\mathbf{S}$ with respect to a time parameter t, $\mathbf{N}$ is an unit surface normal vector, and a speed function F is determined by the Euler-Lagrange equation. Suppose this differential equation have a solution, it is an objective surface. The evolution process is called a surface flow whose each time solution surface is called an evolving surface. The following Figures 1.1 represent an example of the discrete surface flow for each time steps.

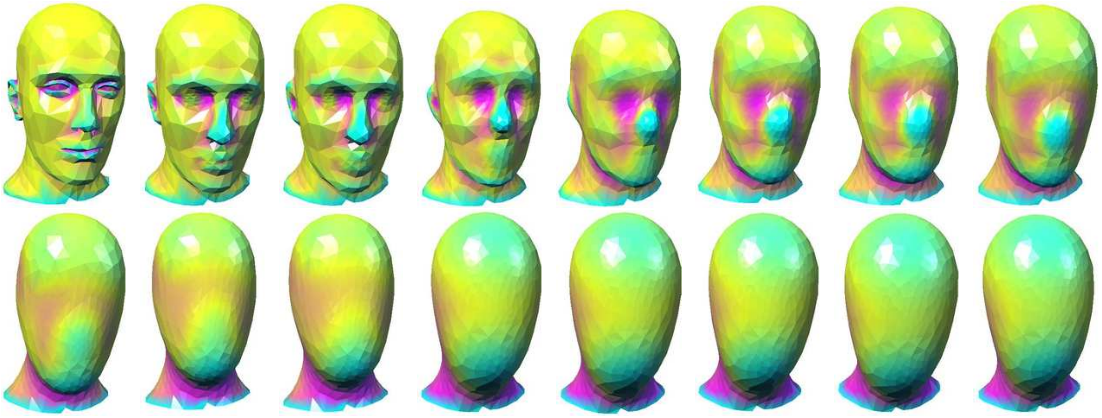


Figure 1.1: Dynamic Meshes via Elastica Flow

1.3 Contributions

The thesis has the following new methods for the triangle mesh modeling via discrete surface evolutions to the shape smoothing, regularization, restoration, geometric feature analysis, and modeling.

Bi-Laplace-Beltrami Flow: The bi-Laplace-Beltrami flow is defined in Section 2.2.6 which is given by the evolutionary system of the partial differential equation

$$\mathbf{S}_t = -\triangle_B^2(\mathbf{S}) \tag{1.2}$$

for the approximation of the elastica flow in Section 2.2.3 derived from consideration of the thin plate energy. Two types of approximations of discrete bi-Laplace-Beltrami flow are implemented in Section 4.3.3.

Discrete Elastica Flow: The new discrete implementation of the elastica flow is developed in Section 4.3. The elastica flow is defined by the evolutionary system of the nonlinear partial differential equation

$$\mathbf{S}_t = -(\triangle_B(H) + 2H(H^2 - K))\mathbf{N}. \tag{1.3}$$

Curvature Visualization on Surface via Pastel and Bipolar Color Mappings: In Section 3.4, the Z-Score thresholding and the Pastel and Bipolar color mapping methods are applied to visualize discrete curvatures on the triangle meshes. This method is expected to use for the feature extraction and recognition of the triangle meshes.

Gaussian Curvature Approximation via Laplace-Beltrami of Normal Method: The Gaussian curvature is approximated by a solution of the simultaneous system of the equation (3.30) in Section 3.5.1. The advantage of the method is that the mean and Gaussian curvatures can be calculated by an only Laplace-Beltrami operator.

Gaussian Curvature Approximation via Modified Desbrun Method: In Section 3.5.1, the modified Desbrun method is given by improving the area weighted angle deficit method to avoid the bad scale of the mean curvature and Gaussian curvature by scaling the Gaussian curvature to satisfy the condition $H^2 - K \geq 0$ without saddle point ($H = 0,\ K \neq 0$). This method is the best method to approximate the mean and Gaussian curvatures for the irregular triangle meshes in the six approximation methods shown with Section 3.5 from the numerical experiments in Section 3.5.5.

Discrete Stable Surface Smoothing via Area Weighted Laplace-Beltrami Flow: According to my experiments [41], the area weighted Laplace-Beltrami operator is applied to the stable surface smoothing which combines the excellent stability properties of the umbrella operator with remarkable geometric properties of the mean curvature normal vector in section 4.1. The method is expected to apply for the feature preserving shape smooting applications, although the method has an automatic iteration control problem.

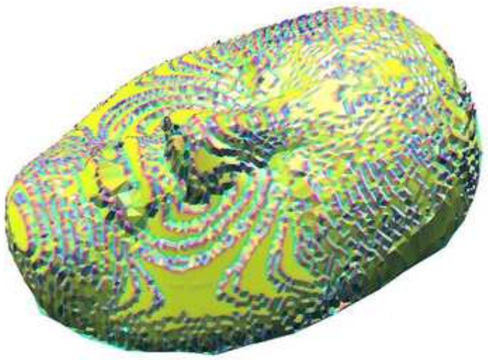


Figure 1.2: Real-World Noh-Mask

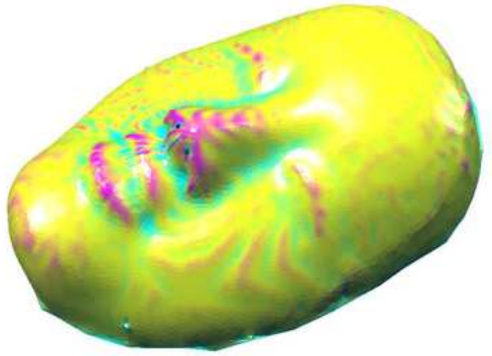


Figure 1.3: Denoising Noh-Mask

Boundary Regularization and Fairing via Weighted Bi-Laplacian Operator: The weighted bi-laplacian tangent flow is developed in Section 4.2.8 to uniformize the boundary vertices for the same geometry parts. The weighted bi-laplacian flow is developed in Section 4.3.1 to fair the boundary vertices.

Mesh Regularization via Area Weighted Bi-Laplace-Beltrami Tangent Flow: The mesh regularization method is implemented in Section 4.2 via the area weighted bi-Laplace-Beltrami tangent flow to uniformize the vertices for the same geometry parts. Although the convergence speed of the proposed new tangent flow is slow speed, the method is useful for the mesh regularization.

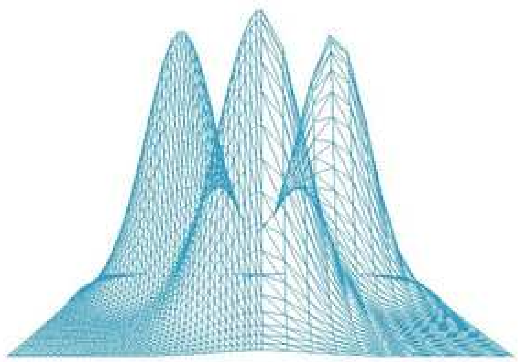


Figure 1.4: Original Irregular Mesh

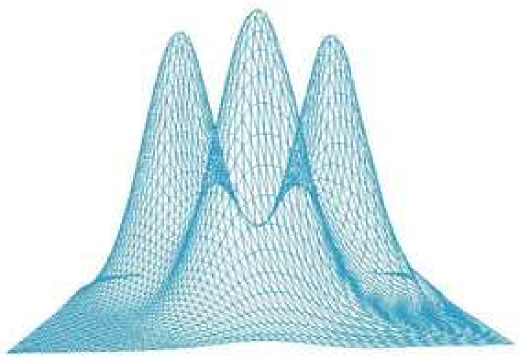


Figure 1.5: Regularized Mesh

Figure 1.4 represents an original irregular mesh model. Figure 1.5 represents the regularized triangle mesh by the proposed new tangent flow.

Fair Surface Modeling via Elastica Flow: In Section 4.3.5, the fair surface modeling is developed by new discrete elastica flow via the curvature approximation method proposed in Section 3.5.1. The limit surface has very good qualities of triangle meshes with interpolated the boundary conditions. The method is expected to develop for the shape restoration modeling, although the convergence speed of the proposed new discrete elastica flow is slow speed.

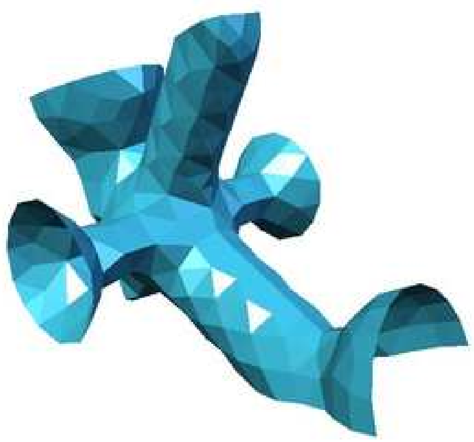


Figure 1.6: Original Water Pipe Model

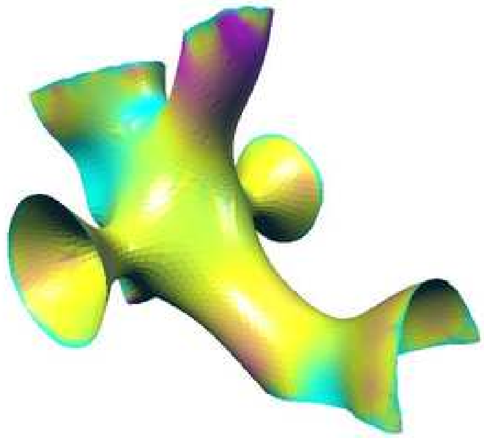


Figure 1.7: Fairing Water Pipe Model via Elastica Flow

Figure 1.6 represents an original water pipe model. Figure 1.7 represents the fairing water pipe with interpolated boundary conditions via the proposed new discrete elastica flow.

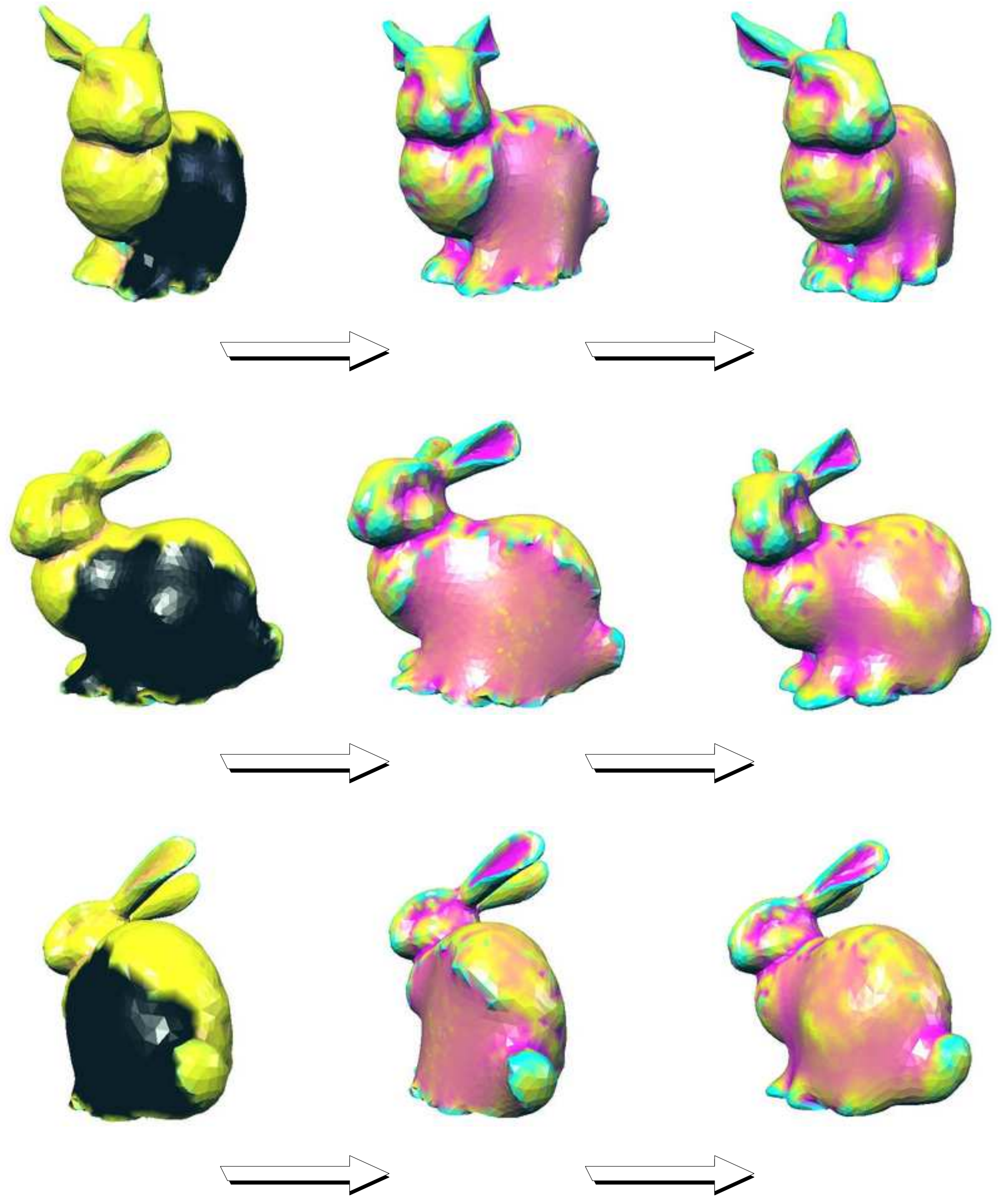


Figure 1.8: Fill Hole with Fairing Bunny

The left-hand side of Figure 1.8 represents a bunny surface with a hole which is confirmed by the black color. The center of Figure 1.8 is the bunny with a reconstructed hole by the flatness plate (ex. Delaunay triangulation). The right-hand side of Figure 1.8 represents the fairing surface with interpolated boundary conditions via the proposed new discrete elastica flow. The method is expected to solve the zipping problem of constructive solid geometry .

Program Implementation: All methods were implemented by Java Development TooKit 1.3 and Java3D application programming interface developed by Sun Microsystems. A few mathematical analysis was calculated by Mathematica 4.0 developed by Wolfram Research. These programs were executed on IRIX 6.5 O2 and Solaris 2.6 Ultra workstations developed by Silicon Graphics and Sun Microsystems, respectively.

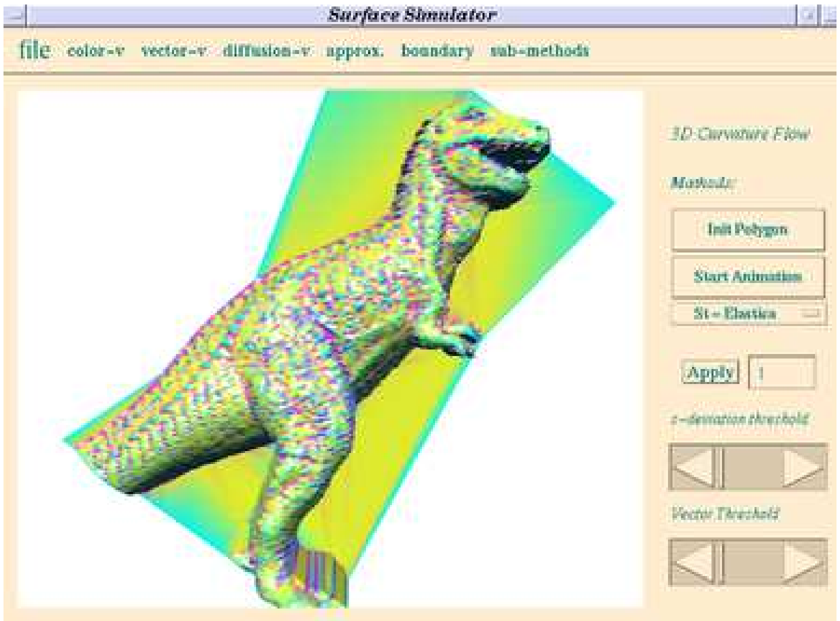


Figure 1.9: Surface Flow Simulator

1.4 Paper Organization

The thesis is organized as follows: four chapters which include this introduction Chapter 1 and three appendices. The variational surface calculus is introduced in Chapter 2. A few analytical surface energies and their Euler-Lagrange equations are considered in Section 2.1. The curvature-driven surface evolution techniques and a few surface flows are introduced in Section 2.2. Analysis of the surface singularities are considered in Section 2.3. The discrete approximation methods of the surface intrinsic properties are developed in Chapter 3. In Section 3.1, the discrete local manifold is defined by the neighborhood triangles of a vertex. The discrete differential operators which are the different types of the Laplace-Beltrami operator are introduced in Section 3.3. Two types of the visualization methods are represented in Section 3.4. The curvature approximation methods are described in Section 3.5. The multi-grid approach and polygonal singularity problem are introduced in Section 3.6 and 3.7, respectively . Three surface modeling applications are implemented in Chapter 4. The stable discrete surface smoothing is developed in Section 4.1. The mesh regularization is achieved in Section 4.2. The fair surface modeling is applied in Section 4.3.

Chapter 2

Elements of Variational Surface Calculus

The chapter discusses the variational surface calculus based on the curvature-driven surface evolution techniques [1, 40]. The surface shapes are modeled by the solutions of the partial differential equations which are derived from minimizing the surface energy functionals. The surface energy functional is an integral equation which is constructed by the function of the geometric properties. The integrand is determined by the desirable physical condition for example elastic equilibrium. The partial differential equation which is equivalent to minimize the functional is called the Euler-Lagrange equation [33] corresponding on the functional. The modeled surface is the solution of the Euler-Lagrange equation. The curvature-driven surface evolution technique solves the Euler-Lagrange equation by the surface flow [1, 17, 50]. The considered energy functionals include the membrane (squared gradient surface), surface bending (squared normal curvature) energies and its approximations.

The geometric variational concepts and curvature-driven surface evolution techniques are often used in computer graphics, shape modeling and analysis [21, 45, 46, 47], image processing and fluid mechanics, shape and image smoothing [11, 13, 20, 31, 41, 43, 48], fair shape modeling [16, 26, 31, 36, 37, 38], optimization [6], curve and surface interpolation [1, 40], grid generation [17], feature extraction and recognition of shape and image [1, 17, 35], and many other applications. The surface flows describe not only the static shapes but also the dynamic shapes for interfaces [9, 17, 50] which are the boundary surfaces of physical phenomenon. In the thesis, the minimizing membrane and elastic bending energies are devoted to the shape denoising, simplification and mesh regularization and fair surface modeling, respectively.

There are many mathematical methods to solve the differential equations which are indicated in Section 2.1 without the surface evolutions method. In [9, 17, 50], the level set method was applied to solve the mean curvature and derivative of the mean curvature flows whose solutions form the minimal surfaces. The level set method is constructed via the implicit function flows which have the excellent advantage about tracing mean curvature flow solutions after developing singularities of the evolving surface. However the high-order derivative flow (ex. elastica flow) is unstable and difficult to drive the evolution equations of the level set method. In [22], the gradient decent method was applied to solve arbitrary surface energies with constraints. The gradient decent method was used in the elastic net algorithm [25] to approximate the Euclidean traveling salesman problem. In [37], the Laplace-Beltrami of mean curvature flow was calculated by the linearization of the directional curvatures and the separation of the fourth order differential equation into two second order differential equations.

The chapter represents surface energies and flows which include second and forth order nonlinear diffusion equations for the surface smoothing and fairing via the geometric variational concepts with curvature-driven surface evolution techniques.

The chapter is organized as follows. A few analytical surface energies and their Euler-Lagrange equations are considered in Section 2.1. The curvature-driven surface evolution techniques and a few surface flows are introduced in Section 2.2. Analysis of the surface singularities are considered in Section 2.3.

2.1 Surface Energy Functionals

Let us introduce the first differential parameter and Laplacian operator on 2D manifold (M^2 metric). Here u and v are curvilinear coordinates on a surface $\mathbf{S}$, E, F and G are the coefficients of the first fundamental forms [5, 10, 34].

The Laplace-Beltrami operator $\triangle_B$ for a function ϕ defined on $\mathbf{S}$ is given by

$$\triangle_B(\phi) \quad = \quad \frac{1}{\sqrt{EG-F^2}}\left(\frac{\partial}{\partial u}\frac{G\phi_u - F\phi_v}{\sqrt{EG-F^2}} + \frac{\partial}{\partial v}\frac{E\phi_v - F\phi_u}{\sqrt{EG-F^2}}\right) \tag{2.1}$$

in [8] page 160. If the metric satisfies the following equation (flat metric)

$$F = 0, \quad E = G = 1, \tag{2.2}$$

then the standard Laplacian $\triangle$ is represented by

$$\triangle(\phi) \quad = \quad \phi_{uu} + \phi_{vv}. \tag{2.3}$$

The first differential parameter $\nabla(\varphi, \psi)$ for the functions φ and ψ is defined by

$$\nabla(\varphi, \psi) \quad = \quad \frac{E\varphi_v\psi_v - F(\varphi_u\psi_v + \varphi_v\psi_u) + G\varphi_u\psi_u}{EG-F^2} \tag{2.4}$$

in [8] page 160. If the metric satisfies the equation (2.2) then this differential parameter represents

$$\nabla(\varphi, \psi) \quad = \quad \varphi_u\psi_v + \varphi_v\psi_u = \mathrm{grad}(\varphi)\cdot\mathrm{grad}(\psi), \quad \mathrm{grad}(\varphi) = (\varphi_u, \varphi_v), \tag{2.5}$$

$$\nabla(\varphi, \varphi) \quad = \quad \varphi_u^2 + \varphi_v^2 = |\mathrm{grad}(\varphi)|^2. \tag{2.6}$$

This operators $\triangle_B()$ and $\nabla(,)$ do not depend on the parametrization of the surface metric but operator $\triangle()$ and $\mathrm{grad}()$ depend on the parametrization of the surface metric.

2.1.1 Membrane Energy

The membrane energy $E_{Membrane}$ on M^2 metric is defined by

$$E_{Membrane} \quad = \quad \iint_\Omega \nabla(\mathbf{S}, \mathbf{S})\, dA = 2 \iint_\Omega dA, \tag{2.7}$$

where Ω is a parameter domain, A is the surface area. The above equation (2.7) shows that minimizing membrane energy on M^2 metric is equivalent to find the minimal surface [5, 30] which satisfies the following Euler-Lagrange equation (2.8), [5, 30, 33].

$$A \Longrightarrow min, \quad \Longleftrightarrow \quad H = 0, \tag{2.8}$$

where H is a surface mean curvature. The minimal surfaces represent the membrane shapes for example the blow soaps, etc.

On Plane: Instead of considering a minimal surface, the minimizing membrane energy of an element represented as a graph of a function $\mathbf{S}(u, v) = (u, v, f(u, v))$ [39, 31] is defined on the flat metric (2.2) by

$$\iint_\Omega |\text{grad}(f)|^2\, dudv \Longrightarrow min. \tag{2.9}$$

The corresponding Euler-Lagrange equation is the Laplace equation [29, 33, 39]:

$$\triangle(f) = 0. \tag{2.10}$$

2.1.2 Elastic Bending Energy

The surface bending energy $E_{bending}$ is defined by

$$E_{bending} = \iint_\Omega \int_0^{2\pi} k_n^2(\theta)\, d\theta\, dA = \iint_\Omega (3H^2 - K)\pi\, dA, \tag{2.11}$$

where $k_n(\theta)$ is a surface normal curvature and K is a Gaussian curvature. The minimizing bending energy $E_{bending} \Longrightarrow min$ is equivalent to find the elastica surface which is a 3D version of elastica curve in [24] and the following variational problems:

$$\iint_\Omega H^2\, dA \quad \Longrightarrow \quad min, \tag{2.12}$$

$$\iint_\Omega (k_{max}^2 + k_{min}^2)\, dA \quad \Longrightarrow \quad min, \tag{2.13}$$

$$\iint_\Omega |\triangle_B(\mathbf{S})|^2\, dA \quad \Longrightarrow \quad min, \tag{2.14}$$

where k_{max} and k_{min} are the surface principal curvatures. Because their Euler-Lagrange equations are equal to the Euler-Lagrange equation of the equation (2.11), (see Appendix A and [33]):

$$\triangle_B(H) + 2H(H^2 - K) = 0. \tag{2.15}$$

The equation (2.14) is equivalent to the constant proportion of the equation (2.12) because of $\triangle_B(\mathbf{S}) = 2H\mathbf{N}$. The double surface integral of the Gaussian curvature is a constant value because of the Gauss-Bonnet theorem [8, 29]. Therefore the equation (2.12), (2.13) and (2.14) give us the same Euler-Lagrange equation.

The bending energy is available for the surface fairness estimation [25, 27, 31, 36]. Because the bending energy preserves and propagates fairness of the boundaries or prescribed vertices to their neighborhood then the elastica surface which satisfies the equation (2.15) have G^2 continuity. The membrane energy does not have such kind of the high continuity around the boundary or prescribed vertex.

A few approximations of the bending energy are considered in [26, 37].

Linear Mean Curvature Distribution Surface: Instead of the nonlinear equation (2.15), the linearized Euler-Lagrange equation is given by

$$\triangle_B(H) = 0. \tag{2.16}$$

Thin Plate Energy: Another possible approximation of the bending energy is the thin plate energy [18, 21, 31, 44] on the flat metric (2.2) which is defined by

$$E_{thin\ plate} = \iint_\Omega |\triangle(f)|^2\, dudv \Longrightarrow min, \tag{2.17}$$

where f is a graph of a function $\mathbf{S}(u, v) = (u, v, f(u, v))$ because M^2 metric thin plate energy is equivalent to the bending energy. The minimizing thin plate energy $E_{thin\ plate} \Longrightarrow min$ is equivalent to find the surface (thin plate) which satisfies the following biharmonic (bi-Laplace) Euler-Lagrange equation (2.18)

$$\triangle^2(f) = 0. \tag{2.18}$$

2.2 Curvature-Driven Surface Evolutions

The curvature-driven surface evolution method [1, 40, 41, 48] is applied to solve the indicated Euler-Lagrange equations (2.8), (2.10), (2.15), (2.16) and (2.18) in the thesis. Mathematically curvature-driven surface evolutions are described by an evolutionary system of nonlinear partial differential equation:

$$\frac{\partial \mathbf{S}(u, v, t)}{\partial t} = F(t, \mathbf{I}, \mathbf{II}, ...)\mathbf{N}(u, v, t), \quad \mathbf{S}(u, v, 0) = \mathbf{S}^0(u, v), \tag{2.19}$$

where u and v parameterize the evolving surface $\mathbf{S}(u, v, t)$, t parameterizes the surface family, $\mathbf{N}$ is an unit surface normal vector, $\mathbf{S}^0(u, v)$ is an initial surface, and $F()$ is called a speed function whose arguments are the geometric surface properties which does not depend on the surface parametrization (ex. $\mathbf{I}$ and $\mathbf{II}$ are the first and second fundamental form of the surface).

Suppose the evolving surface satisfies the prescribed boundary conditions (closed surface case), and converges to a limit surface $\mathbf{S}(u, v, \infty)$ where $t \to \infty$. Where the evolution equation (2.19) does not change the limit surface $\mathbf{S}(u, v, \infty)$, the limit surface is a stationary solution of the equation (2.19), then $F(\infty, \mathbf{I}, \mathbf{II}, ...) \equiv 0$ along $\mathbf{S}(u, v, \infty)$. Therefore the curvature-driven surface evolution method can be used, and the Euler-Lagrange equation can be took as the speed function $F()$ for finding the objective surface which satisfies the Euler-Lagrange equation. Depending on the speed function the different flow methods (see equation (2.22), (2.24), (2.25) and (2.27)) are defined. Instead of the notation $\frac{\partial \mathbf{S}(u,v,t)}{\partial t}$, $\mathbf{S}_t$ denotes the time derivative of the surface $\mathbf{S}(u, v, t)$ in the thesis.

Flow Direction: The sign of the speed function is determined by the highest order of the derivative in the arguments of $F()$ of equation (2.19). If highest order of the derivative is a second order, the speed function is equal to the Euler-Lagrange equation from consideration of the heat equation problem [17]. For the fourth order derivative, let us consider the Fourier transform of a function $f(x)$:

$$f(x) \sim \tilde{f}(\xi) = \int_{-\infty}^{\infty} e^{-ix\xi} f(x)\, dx,$$

where x is a parameter and i is an imaginary number $\sqrt{-1}$. The second order derivative (Laplacian) of $f(x)$ is given by

$$\triangle(f(x)) \sim -|\xi|^2 \tilde{f}(\xi). \tag{2.20}$$

The fourth order derivative (bi-Laplacian) of $f(x)$ is given by

$$\triangle^2(f(x)) \sim |\xi|^4 \tilde{f}(\xi). \tag{2.21}$$

The sign of the bi-Laplacian (2.21) is inverse of the Laplacian (2.20). Therefore if the highest order of the derivative in the arguments is fourth order then the speed function $F()$ takes the minus Euler-Lagrange equation. The dissipative flows, operators, evolutions suppress high frequences of a function then these effect smoothing of a function.

2.2.1 Mean Curvature Flow

The mean curvature flow is given by the evolutionary system of the quasi-linear partial differential equation

$$\mathbf{S}_t = H\mathbf{N}, \tag{2.22}$$

where H is the surface mean curvature and $\mathbf{N}$ is the surface unit normal vector. The evolving surface converges to the minimal surface. The mean curvature flow has the smoothing effect for the surface, because this flow is a shape deformations which has the maximum speed of decreasing surface area in an arbitrary shape deformations. Therefore this flow was applied to the shape denoising and simplification in the several works [17, 31, 35, 41, 48, 50], and this thesis.

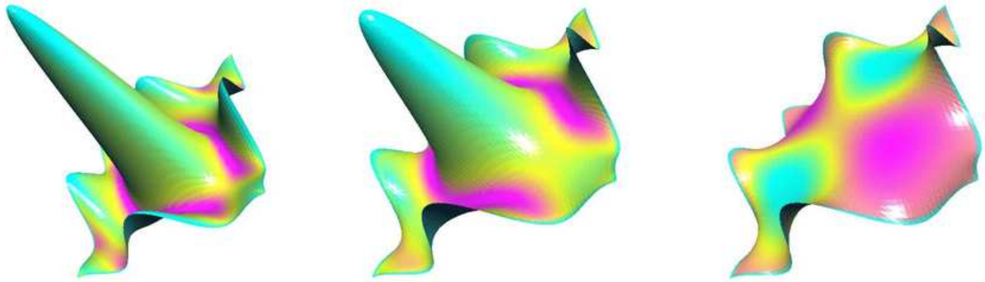


Figure 2.1: Evolving Surfaces via Mean Curvature Flow with Fixed Boundary

The left figure represents an initial surface. The center figure shows an evolving surface during the mean curvature flow. The left figure indicates a limit surface via the mean curvature flow.

The evolving surface will be vanished in finite time when the surface is a closed surface. For example, a sphere converges to one point (see Section 2.3) via this flow. The evolving surface converges to the minimal surface where the surface has the boundaries or prescribed vertices. Here the minimal surface is the surface which has the minimum surface area in an arbitrary surface which satisfies the given boundary conditions. Regrettably, the evolving surface is not necessarily always avoiding surface singularities during evolutions (see Section 2.3 and [7, 17, 50]) before vanishing.

2.2.2 Heat Equation

The simplified heat equation is given by the evolutionary system of the linear partial differential equation

$$f_t = \triangle(f) \tag{2.23}$$

to solve the Laplace equation (2.10) for approximation of the mean curvature flow [17, 20, 26, 31, 41, 48] in the above Section 2.2.1. This popular flow describes a propagation of the thermal diffusion which represents the smoothing effect of a family of the isothermal surfaces (evolving surfaces of the evolution) because this flow corresponds a low-pass filtering of the Fourier analysis considered in many text books of the differential equations, thermodynamics, Fourier analysis and so on. The heat equation (2.23) has an excellent stability for numerical implementations because of their linear characteristics, and a fundamental solution existence. However this flow depends on the surface (function) parametrization.

2.2.3 Elastica Flow

The elastica flow is given by the evolutionary system of the nonlinear partial differential equation

$$\mathbf{S}_t = -(\triangle_B(H) + 2H(H^2 - K))\mathbf{N}, \tag{2.24}$$

where K is a surface Gaussian curvature. The evolving surface converges to an elastica surface [25] which has the minimizing the surface bending energy functional. It means that the limit surface has G^2 continuity with minimum bending energy. Two dimensional version of the elastica surface is called the elastica curve [24, 34] or mechanical spline [1, 10, 40]. The elastica flow has the smoothing effect for the surface. The elastica flow may preserve the volume bounded by the closed surface, because the mechanical spline preserves the area bounded by the closed curve.

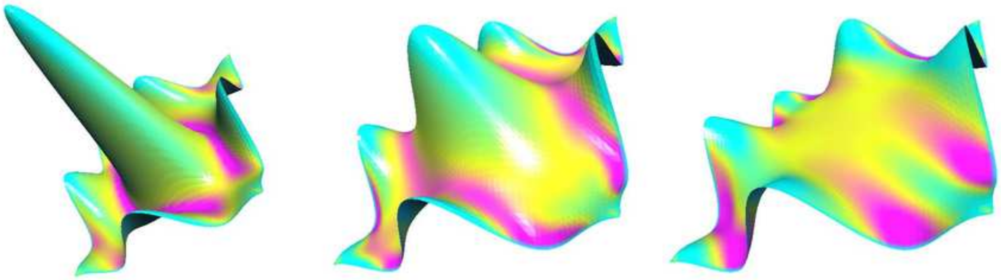


Figure 2.2: Evolving Surfaces via Elastica Flow with Fixed Boundary

The left figure represents an initial surface. The center figure shows an evolving surface during the elastica flow. The left figure indicates a limit surface via the elastica flow.

The evolving surface does not be vanished [25] in finite time, when the surface is a closed surface. For example, a ellipse converges to the sphere via this flow. Also the evolving surface converges to the closed elastica surface which is characterized by the topology of the initial surface [25]. Here the surface is an open surface then the evolving surface converge to the elastica surface which interpolates the prescribed boundary conditions (boundary curvatures, normal directions, tangent directions, or fixed strips) with fixed vertices to satisfy G^2 continuity with minimum bending energy. The evolving surface may not develop surface singularities during evolutions (see Section 2.3 and [25]). This flow is applied to the fair surface modeling in [16, 26, 31, 36, 37, 38] and this thesis from consideration of the above reasons.

2.2.4 Laplace-Beltrami of Mean Curvature Flow

The Laplace-Beltrami of mean curvature flow (**LBoMC**) is given by the evolutionary system of the quasi-linear partial differential equation

$$\mathbf{S}_t = -\triangle_B(H)\mathbf{N} \tag{2.25}$$

to solve the equation (2.16) for the approximation of the elastica flow in the above Section 2.2.3 via the linearization of the equation (2.15). This flow was used in [38] for fair surface modeling purposes. This flow develops surface singularities during evolutions (see Section 2.3). The numerical stability is better than the elastica flow because of their quasi-linear form of the speed function.

The limit surface interpolates the surface boundaries where the surface has the boundaries or prescribed vertices and the linear distribution of the mean curvature. Regrettably, the evolving surface develops surface singularities during evolutions in the same manner of the mean curvature flow, there are no mathematical analysis of the general closed surface [50].

2.2.5 Biharmonic Diffusion Equation

The biharmonic diffusion equation is given by the evolutionary system of the biharmonic partial differential equation

$$f_t = -\triangle^2(f) \tag{2.26}$$

to find the solution of the equation (2.18) whose limit surface has the minimum thin plate energy [11, 13, 31] on the flat metric (2.2). The evolving surface will grow to propagate the Laplacian of f. This flow is a stable evolution because of their linear characteristics. However this flow depends on the surface parametrization in the same manner of Section 2.2.2.

2.2.6 Bi-Laplace-Beltrami Flow

The bi-Laplace-Beltrami flow is given by the evolutionary system of the quasi-linear partial differential equation

$$\mathbf{S}_t = -\triangle_B^2(\mathbf{S}) \tag{2.27}$$

for the approximation of the elastica flow in Section 2.2.3 derived from consideration of the thin plate energy. This is a new flow to approximate minimizing the surface energy functional. This flow does not depend on the surface parametrization. Also the numerical stability is better than the elastica flow and **LBoMC** flow because of their speed function. The speed function sign is always same sign of the **LBoMC** flow about a same surface from the Fourier transform of the Laplacian which was represented in the equation (2.20) and (2.21).

The limit surface interpolates the surface boundaries where the surface has the boundaries or prescribed vertices. Regrettably, the evolving surface develops surface singularities during evolutions in the same manner of the mean curvature flow.

2.3 Singularities Developed by Surface Flows

An evolving surface may develop singularities during evolutions [7, 17, 50] before vanishing without the elastica flow [25]. The curve curvature flow does not develop singularities [50] before vanishing. The analytic surface singularity vertex is a vertex which satisfies the following condition

$$EG - F^2 < 0 \Longleftrightarrow H \to \infty, \quad K \to \infty. \tag{2.28}$$

Before analysis for the function of the curvature flows, let us consider the constant flow case. The normal vector direction is defined by a positive sign of the curvature in the following considerations.

Constant Flow: The constant flow is given by

$$\mathbf{S}_t = \mathbf{N} \tag{2.29}$$

which is equivalent to the d-offset surface [34], where $0 \le d \le \lambda$. The parameter λ is a maximum absolute value of the $\frac{1}{k_{max}}$ or $\frac{1}{k_{min}}$. This flow develops the singularities [3, 17, 50] on the evolute cusps [2, 3, 4, 42], where d takes λ. Because the envelop of a family of the surface normals is the evolute surface, therefore surface normals are intersected each other at the evolute cusps, where d takes λ.

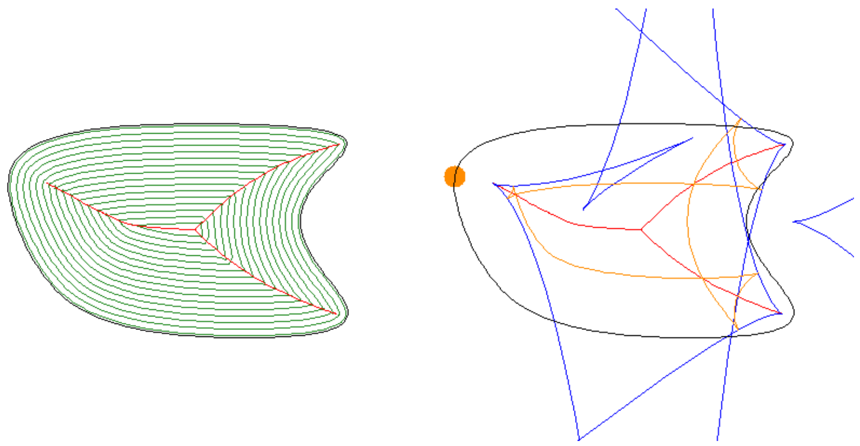


Figure 2.3: 2D Constant Flow with Singularities

The left figure represents the planer level sets with their singularities (skeleton). The right figure shows that the offset and evolute curves describe the singularities of the 2D constant flow. In 3D case, the singularities are developed on the skeletal sheets.

Let us analyze that the surface flows develop the singularities on at least one surface among three simple surface models except the elastica flow.

2.3.1 Sphere

The direction of the normal vector of the sphere is oriented from the center of the sphere to the sphere. The principal curvatures of the sphere whose radius is $r(t)$ are given by $k_{max} = k_{min} = -\frac{1}{r(t)}$. Therefore, $H = -\frac{1}{r(t)}$, $K = \frac{1}{r^2(t)}$ and $\triangle_B(H) = 0$.

Mean Curvature Flow: The speed function is given by $-\frac{1}{r(t)}$ from equation (2.22), therefore the evolving surface converges to the center of the sphere.

LBoMC Flow: The speed function is equal to zero from equation (2.25), therefore the evolving surface does not move from the initial sphere.

Bi-Laplace-Beltrami Flow: The speed function is given by $-\frac{4}{r^3(t)}$ from equation (2.27), therefore the evolving surface converges to the center of the sphere.

Elastica Flow: The speed function is equal to zero from equation (2.24), therefore the evolving surface does not move from the initial sphere.

2.3.2 Cylinder

Let us consider the infinit cylinder. The radius of the cylinder is $r(t)$. The direction of the normal vector of the cylinder is oriented from the center line of the cylinder to the cylinder. The principal curvatures of the cylinder whose radius is $r(t)$ are given by $k_{max} = 0$ and $k_{min} = -\frac{1}{r(t)}$. Therefore, $H = -\frac{1}{2r(t)}$, $K = 0$ and $\triangle_B(H) = 0$.

Mean Curvature Flow: The speed function is given by $-\frac{1}{2r(t)}$ from equation (2.22), therefore the evolving surface converges to the center line of the cylinder.

LBoMC Flow: The speed function is equal to zero from equation (2.25), therefore the evolving surface does not move from the initial cylinder.

Bi-Laplace-Beltrami Flow: The speed function is given by $-\frac{1}{r^3(t)}$ from equation (2.27), therefore the evolving surface converges to the center line of the cylinder.

Elastica Flow: The speed function is given by $\frac{1}{4r^3(t)}$ from equation (2.24), therefore the evolving surface diverges by same speeds at all vertices on the cylinder.

2.3.3 Barbell

Let us show the local shape analysis of the **LBoMC** to estimate the flow directions of the barbell shape.

Local Shape Analysis of LBoMC: The Taylor expansion of the surface is given by

$$\begin{aligned}\mathbf{S}(u,v) &= \frac{1}{2}(k_1u^2 + k_2v^2) + \\ &+\frac{1}{6}(a_1u^3 + a_2u^2v + a_3uv^2 + a_4v^3) + \\ &+\frac{1}{24}(b_1u^4 + b_2u^3v + b_3u^2v^2 + b_4uv^3 + b_5v^4) + \\ &+\frac{1}{120}(c_1u^5 + c_2u^4v + c_3u^3v^2 + c_4u^2v^3 + c_5uv^4 + c_6v^5) + O(u^6, v^6),\end{aligned}$$

where u and v are the principal parameterization (du and dv give the principal directions), k_1 and k_2 are the principal curvatures ($H = \frac{1}{2}(k_1+k_2)$ and $K = k_1k_2$), a_i, b_i and c_i are the constant coefficients, and $O(u^6, v^6)$ is the high-order terms. The **LBoMC** of above equation is given by

$$\begin{aligned}\triangle_B(H)\,|_{u=v=0} &= -(k_1^3 + k_2^3) + \frac{1}{6}(3b_1 + b_3 + 3b_5) \\ &= -2H(4H^2 - 5K) + \frac{1}{6}(3b_1 + b_3 + 3b_5) \\ &= -2H(4H^2 - 5K) + \frac{1}{2}\triangle^2(\mathbf{S})|_{u=v=0},\end{aligned}$$

because $\triangle^2(\mathbf{S})|_{u=v=0} = (b_1 + \frac{1}{3}b_3 + b_5)$.

Let us define the barbell shape (see Figure 2.4) which is constructed by three geometry parts:

- **Spherical Part:** The spherical part has the radius $R(t)$, therefore,

$$H_I = -\frac{1}{R(t)}, \quad K_I = \frac{1}{R^2(t)}, \quad -\triangle_B(H_I) = 0,$$

  where H_I and K_I are the mean and Gaussian curvatures of the spherical part, respectively.

- **Cylindrical Part:** The cylindrical part has the radius $r(t)$, therefore,

$$H_{II} = -\frac{1}{2r(t)}, \quad K_{II} = 0, \quad -\triangle_B(H_{II}) = 0,$$

  where H_{II} and K_{II} are the mean and Gaussian curvatures of the cylindrical part, respectively.

- **Connection Part:** From the Fourier transform of the Laplacian, the direction of $-\triangle^2(\mathbf{S})|_{u=v=0}$ is always coincidence with the surface normal direction. Therefore, the connection part has the condition

$$H_{III} > 0, \quad K_{III} < 0,$$

$$-\triangle_B(H_{III}) \approx 2H_{III}(4H_{III}^2 - 5K_{III}) - \frac{1}{2}\triangle^2(\mathbf{S})|_{u=v=0} > 0,$$

  where H_{III} and K_{III} are the mean and Gaussian curvatures of the connection part, respectively.

There are relationships:

$$2r(t) << R(t) << h(t) \Longrightarrow |H_{II}| >> |H_I| >> |H_{III}|, \quad K_{III} << K_{II} << K_I.$$

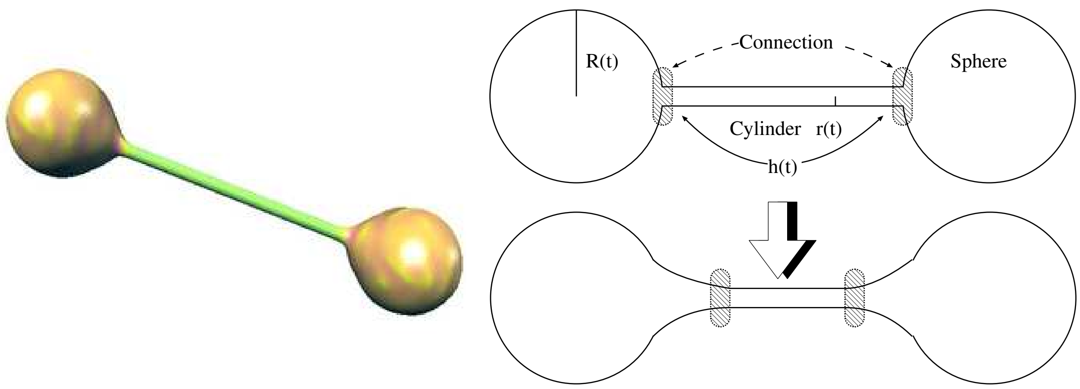


Figure 2.4: Barbell Shape

Mean Curvature Flow: Before the connection part will grow to spherical part (from $H_{III} > 0$), the cylindrical part converges the center line of the cylindrical part because of $|H_{II}| >> |H_I| >> |H_{III}|$. Therefore the mean curvature flow can be develop singularity, see Figure 2.5.

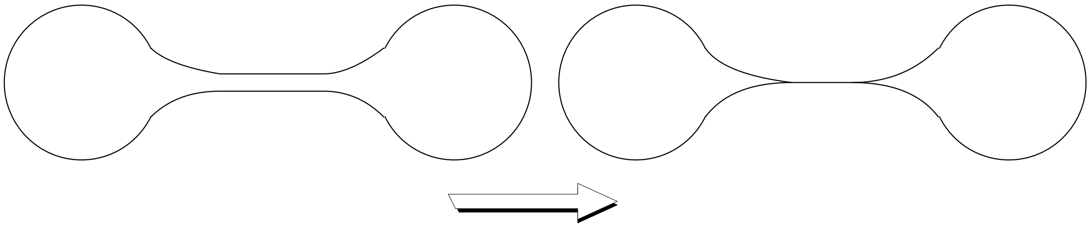


Figure 2.5: Singularities via Mean Curvature Flow

LBoMC Flow: Before the connection part will grow to spherical part ($-\triangle_B(H_{III}) > 0$), the cylindrical part develops two singular points on connection between two cylindrical parts and connection parts. Because the mean curvature of these two parts H satisfy $H < 0$ and $|H_{III}| << |H|$ then $-\triangle_B(H) > 0$, therefore these two parts converge points, see Figure 2.6.

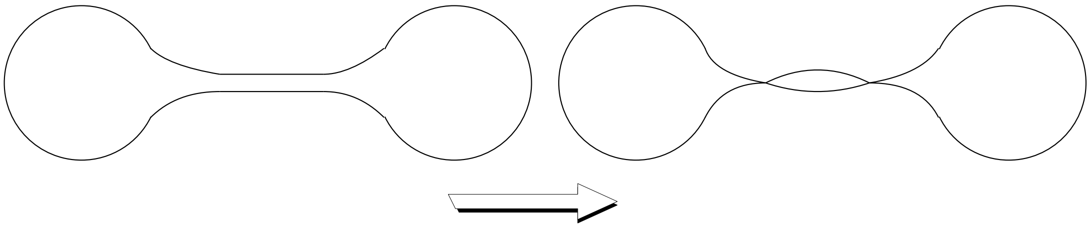


Figure 2.6: Singularities via Laplace-Beltrami of Mean Curvature Flow

Bi-Laplace-Beltrami Flow: The cylindrical part speed is equal to $-\frac{1}{r^3(t)}$ then this part converges very quickly to the center line. The bi-Laplace-Beltrami flow behaves similarly the **LBoMC** flow about connection part. Therefore the mean curvature flow can be develop singularity, see Figure 2.7.

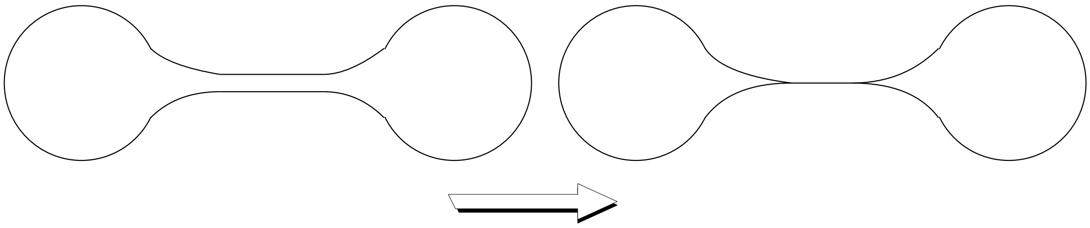


Figure 2.7: Singularities via Bi-Laplace-Beltrami Flow

Elastica Flow: The cylindrical part will grow fat very fast speed because of

$$-(\triangle_B(H_{II}) + 2H_{II}(H_{II}^2 - K_{II})) = \frac{1}{4r^3(t)}.$$

The spherical part does not change while small time steps because of

$$-(\triangle_B(H_I) + 2H_I(H_I^2 - K_I)) = 0.$$

The speed function of the connection part is approximated by

$$\begin{aligned} & -(\triangle_B(H_{III}) + 2H_{III}(H_{III}^2 - K_{III})) \\ \approx \quad & -(-2H_{III}(4H_{III}^2 - 5K_{III}) + 2H_{III}(H_{III}^2 - K_{III})) - \triangle^2(\mathbf{S})|_{u=v=0} > 0, \end{aligned}$$

because of the Fourier transform of the Laplacian in equation (2.20) and $H_{III} > 0$. For all parts of the barbell shape, the signs of the speed function are same direction which is signed for the normal direction. The barbell will grow to the ellipse shape via the elastica flow, see Figure 2.8. Therefore the elastica flow may not develop the singularity where the initial evolving surface is the barbell shape.

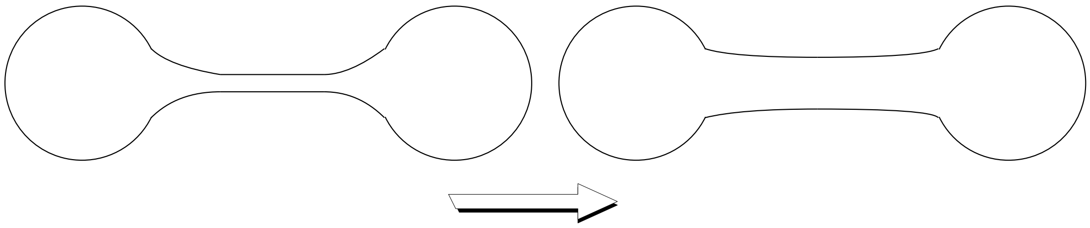


Figure 2.8: Estimated Evolution of Elastica Flow

Chapter 3

Discrete Differential Geometry

The chapter develops the discrete approximations of the geometric surface properties and the numerical methods to solve the surface flows represented in Chapter 2. Since triangle meshes are becoming extremely popular in computer graphics applications for approximation of smooth continuous surfaces with arbitrary shape and topology. The surface shapes are constructed by a set of the vertices and triangle faces with their topology (see Figure 3.1). In Section 3.1, the discrete local manifold is defined by the neighborhood triangles of a vertex. The meshing types are classified by the connectivity (vertex valence) and vertex distribution.

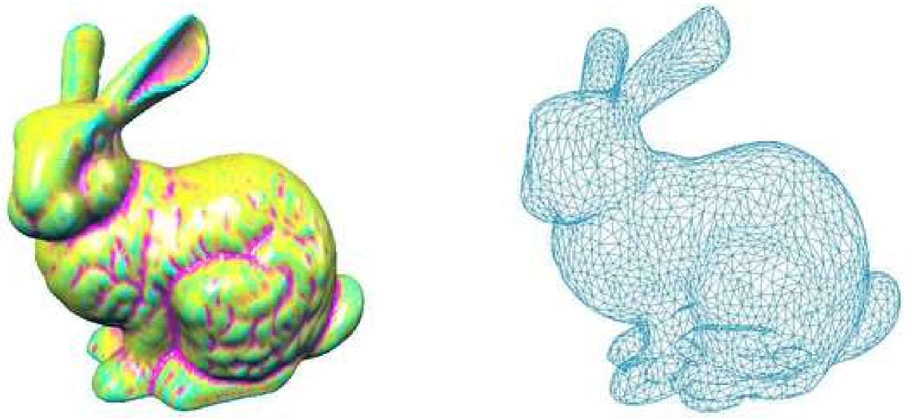


Figure 3.1: Triangulated Polygonal Approximation

The forward Euler difference method [17, 41] is applied to implement the curvature-driven surface evolution techniques in Section 3.2.

The discrete differential operators which are the different types of the Laplace-Beltrami operator are introduced in Section 3.3. A few surface modeling applications were used in [31, 36, 37, 38, 41, 48], Desbrun et al. developed a very useful discrete Laplace-Beltrami operator [31, 32] for the triangle meshes. The most popular Laplacian approximation is an umbrella operator [11, 13, 20, 26, 27, 31, 41, 48] which have the smoothing effect with excellent stability of the surface flow. New approximation of the Laplace-Beltrami operator which have the interim effects between Desbrun's operator and the umbrella operator is proposed.

Many discrete curvature approximation methods were considered in previous works [12, 16, 17, 25, 26, 31, 32, 41, 45, 46, 50]. Two types of the visualization methods are represented in Section 3.4. Four popular and two new curvature approximation methods are described in Section 3.5. The proposed new modified Desbrun method is the best method to approximate the mean and Gaussian curvatures about triangulated polygonal meshes from the numerical experiments in Section 3.5.1.

At the end of the chapter, a multi-grid approach to solve differential equations numerically and polygonal singularity problem are introduced in Section 3.6 and 3.7, respectively.

3.1 Triangulated Polygonal Meshes

Instead of considering a smooth continuous surface, the triangulated polygonal approximation is used in the thesis. Let us define the 1-ring neighborhood at an evaluation vertex $\mathbf{P}$ with notation of their discrete geometric properties for a locally manifold.

1-Ring Neighborhood (Local Manifolds): The 1-ring neighborhood is a set of the triangles which are located around on the evaluation vertex $\mathbf{P}$ whose neighbor vertices $\mathbf{Q}_i$ are connected for $\mathbf{P}$ by the one edge path between $\mathbf{P}$ and $\mathbf{Q}_i$. The following Figure 3.2 represents the 1-ring neighborhood.

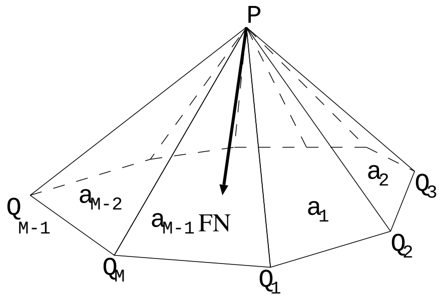


Figure 3.2: 1-Ring Neighborhood

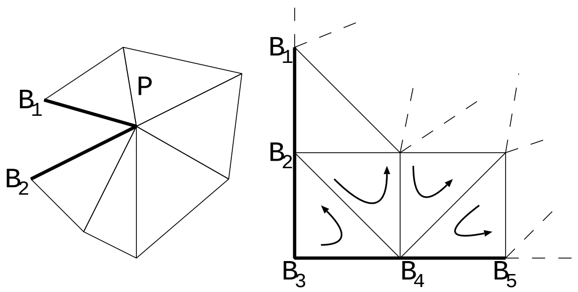


Figure 3.3: Boundary

Manifold with Link: A link of the vertex $\mathbf{P}$ is a set of the edges of 1-ring neighborhood which does not include the vertex $\mathbf{P}$. The vertex $\mathbf{P}$ is a manifold vertex where the link of $\mathbf{P}$ is a closed polygonal curve.

Boundary Vertex: The vertex $\mathbf{P}$ is a boundary vertex where the link of $\mathbf{P}$ is an open polygonal curve. For programming purpose, the boundary vertex is recognized by following definition; If a number of 1-ring neighbor vertices M is not equal to a number of 1-ring neighbor triangles then the evaluation vertex is a boundary vertex (see Figure 3.3). If the triangulated meshes does not have the non-manifold vertex then the boundaries are the set of the closed polylines.

Non-Manifold: The vertex is a non-manifold vertex, where the vertex is not the manifold and boundary vertex. It means that the vertex neighbor can not be recognized by a disc or a half disc. Only manifold data is devoted in the thesis.

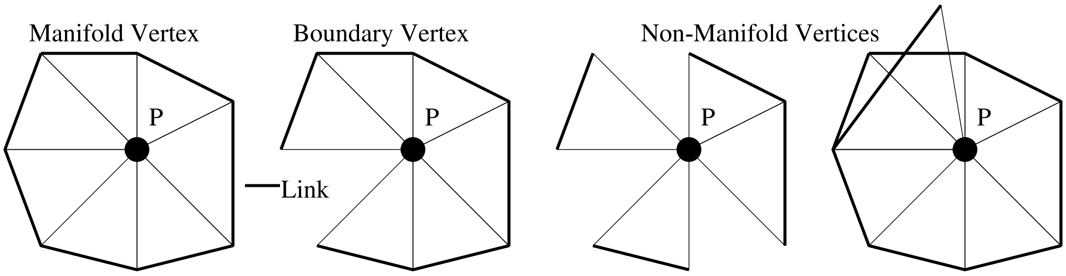


Figure 3.4: Link with Vertex Classification

Boundary Edge: The edge is a boundary edge, if the edge appears only once, when the edges of 1-ring neighborhood are traced along their triangle orientations. For example, when the evaluation vertex is B_4 of Figure 3.3 then the both end points of the edge B_4 and B_2 are the boundary points but the edge B_4B_2 was counted two times, therefore the edge B_4B_2 is not a boundary edge.

Boundary Strip: The boundary strip is a set of the triangles whose triangle has at least one boundary vertex.

Irregular Meshes: The irregular meshes are characterized by the following three irregularities: Irregular Vertex Distribution, Irregular Connectivity Distribution and Irregular Angle Distribution which are the strongly non-uniform vertex, valence and angle distribution for same geometry, respectively.

3.2 Discrete Flow (Dynamic Meshes)

The system of the analytic surface flows are usually solved by either an explicit or implicit numerical scheme [17]. Although the explicit numerical schemes are unstable for large time step-size, they are widely used because their implementation is simple and the nonlinear speed functions can not be implemented to the implicit scheme which have the matrix of simultaneous equations. The evolution equation (2.19) is discretized by the algebraic operation:

$$\mathbf{S}_{new} \leftarrow \mathbf{S}_{old} + \epsilon F_{old}\mathbf{N}_{old}, \quad \epsilon << 1, \tag{3.1}$$

via the forward Euler difference method for the time parameter domain. Old vertices $\mathbf{S}_{old}$ are updated by their speed vectors $F_{old}\mathbf{N}_{old}$ in the positions of a new surface $\mathbf{S}_{new}$, respectively. The old meshes are displaced to new meshes whose desired energy have less than the old one.

3.3 Discrete Differential Operators

Several different approximations of the Laplace operator are introduced for the triangle meshes (see Figure 3.2 and 3.6).

Weighted Umbrella Operator: The Laplace-Beltrami operator is discretized by the weighted umbrella operator $L_w()$ [31, 48]

$$L_w(\mathbf{P}) = \frac{1}{W}\sum_{k=1}^{M} w_i(\mathbf{Q}_i - \mathbf{P}), \tag{3.2}$$

where w_i is a weight of the neighbor vertex $\mathbf{Q}_i$ and W is a weight for the 1-ring neighborhood. The Laplace-Beltrami operator approximations which are defined in this Section can be represented by the above weighted umbrella operator. Usually, weights w_i and W take the polygonal properties for example edge length, angle, etc.

Standard Umbrella Operator: For the approximation of the Laplacian $\triangle()$, the standard umbrella operator $L()$ is given by

$$L(\mathbf{P}) = \frac{1}{M}\sum_{i=1}^{M} \mathbf{Q}_i - \mathbf{P}, \quad w_i = 1, \quad W = M. \tag{3.3}$$

This operator has the smoothing effect with excellent stability of the flow, but the flow strongly depends on the vertex distribution [41]. The smoothing effect between the umbrella operator and Fourier transform were analyzed in [11, 13].

Proposition: *Let us consider a uniform dense triangle mesh approximating well a smooth surface. Then the umbrella operator provides a good approximation of the mean curvature normal vector.*

Justification of Proposition: *Let us expand a surface $f(x, y)$ by Taylor series expansion*

$$\begin{aligned} f(x,y) &= f(0,0) + xf_x(0,0) + yf_y(0,0) + \\ &\quad + \frac{x^2}{2}f_{xx}(0,0) + xyf_{xy}(0,0) + \frac{y^2}{2}f_{yy}(0,0) + O(x^3, y^3), \end{aligned}$$

where $O(x^3, y^3)$ is the high-order term. Integration and division are performed to the both side of above equation as follows.

$$\frac{1}{|C_r|}\int_{C_r} f(x,y)\, ds = f(0,0) + \frac{r^2}{4}\triangle(f(0,0)) + O(r^4),$$

where $O(r^4)$ is the high-order term, r is a radius of local neighborhood of a circle C_r (see Figure 3.5), $|C_r|$ represents the arc length, and s is the arc length parameter. Because,

$$\int_{C_r} x\, ds = \int_{C_r} y\, ds = \int_{C_r} xy\, ds = 0, \quad \frac{1}{|C_r|}\int_{C_r} x^2\, ds = \frac{1}{|C_r|}\int_{C_r} y^2\, ds = \frac{r^2}{2},$$

$$\int_{C_r} x^2\, ds + \int_{C_r} y^2\, ds = \int_{C_r} (x^2 + y^2)\, ds = 2\pi r^3.$$

The Laplacian of the surface is given by

$$\triangle(f(0,0)) \;=\; \lim_{r\to 0} \frac{4}{r}\left[\frac{1}{|C_r|}\int_{C_r} f(x,y)\, ds - f(0,0)\right]. \tag{3.4}$$

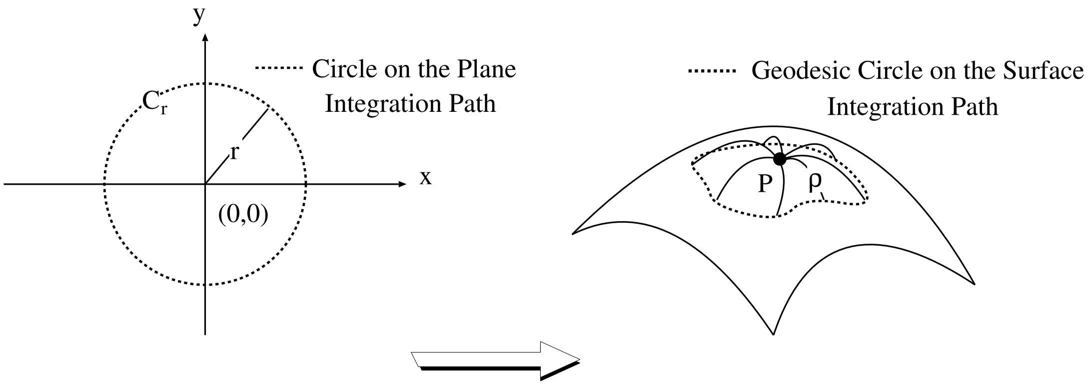


Figure 3.5: Integration Path on Different Metrics

Now let us extend equation (3.4) to the surface case. The coordinates origin is set to a center $\mathbf{P}$ *of the geodesic circle* S_ρ *whose radius* ρ *is a constant distance measured along a geodesic [8], see Figure 3.5. Then, the equation (3.4) can be rewrote by*

$$\triangle_B(f(\mathbf{P})) \;=\; \lim_{\rho\to 0} \frac{4}{\rho^2}\left[\frac{1}{|S_\rho|}\int_{S_\rho} f(x,y)\, ds - f(\mathbf{P})\right], \tag{3.5}$$

where $|S_\rho|$ *represents the arc length on the surface metric* $\mathbf{S}$*. The surface* $\mathbf{S}$ *it self is substituted in the above equation (3.5), then*

$$\triangle_B(\mathbf{S}(\mathbf{P})) \;=\; \lim_{\rho\to 0} \frac{4}{\rho^2}\left[\frac{1}{|S_\rho|}\int_{S_\rho} \mathbf{S}(u,v)\, ds - \mathbf{S}(\mathbf{P})\right] = 2H\mathbf{N}, \tag{3.6}$$

where u *and* v *are the curvilinear coordinates on a surface* $\mathbf{S}$*,* H *is a mean curvature at* $\mathbf{P}$*, and* $\mathbf{N}$ *is an unit normal vector at* $\mathbf{P}$*. Suppose, the smooth surface is approximated by the uniform dense triangle mesh. Also the geodesic circle is considered as the 1-ring neighborhood, then the equation (3.6) can be discretized by the umbrella operator.*

Desbrun's Laplace-Beltrami Operator: The useful approximation of the Laplace-Beltrami operator $\triangle_{B_{Des}}()$ which is developed in [32] is given by

$$\triangle_B(\mathbf{S}) \approx \triangle_{B_{Des}}(\mathbf{P}) = \frac{1}{2A}\sum_{i=1}^{M}(cot\ \alpha_i + \cot\ \beta_i)(\mathbf{Q}_i - \mathbf{P}), \tag{3.7}$$

where α_i and β_i are the angles between $\overrightarrow{\mathbf{Q}_{i-1}\mathbf{P}}$ and $\overrightarrow{\mathbf{Q}_{i-1}\mathbf{Q}_i}$, and $\overrightarrow{\mathbf{Q}_{i+1}\mathbf{P}}$ and $\overrightarrow{\mathbf{Q}_{i+1}\mathbf{Q}_i}$, respectively (see Figure 3.6). This operator provides us better approximation of the mean curvature normal vector than the umbrella operator, where the surface is constructed by the irregular meshes [31, 32, 41]. For the discrete scalar function on the surface (ex. discrete curvatures), this Laplace-Beltrami operator can be defined by

$$\triangle_B(\phi_{\mathbf{S}}) \approx \triangle_{B_{Des}}(\phi_{\mathbf{P}}) = \frac{1}{2A}\sum_{i=1}^{M}(cot\ \alpha_i + \cot\ \beta_i)(\phi_{\mathbf{Q}_i} - \phi_{\mathbf{P}}), \tag{3.8}$$

where $\phi_{\mathbf{S}}$ and $\phi_{\mathbf{P}}$ are the scalar values of ϕ on the smooth surface $\mathbf{S}$ and the vertex $\mathbf{P}$ of the polygonal surface, respectively.

Area Weighted Laplace-Beltrami Operator: Another approximation of the Laplace-Beltrami operator is the area weighted normalized umbrella operator which is the area weighted approximation of the equation (3.6) developed for the discrete mean curvature flow in [41]. The area weighted Laplace-Beltrami operator $\triangle_{B_{Area}}()$ is given by

$$\triangle_B(\mathbf{S}) \approx \triangle_{B_{Area}}(\mathbf{P}) = \frac{1}{2AM}\sum_{i=1}^{M} a_i\left(\frac{\overrightarrow{\mathbf{PQ}_i}}{|\overrightarrow{\mathbf{PQ}_i}|} + \frac{\overrightarrow{\mathbf{PQ}_{i+1}}}{|\overrightarrow{\mathbf{PQ}_{i+1}}|}\right), \tag{3.9}$$

where a_i is the triangle area and A is a summation of a_i (total 1-ring area). According to my experiments [41], this operator has middle effects between the umbrella and Desbrun's operator for the flow stability, speed and mesh (connectivity and vertex distribution) dependency.

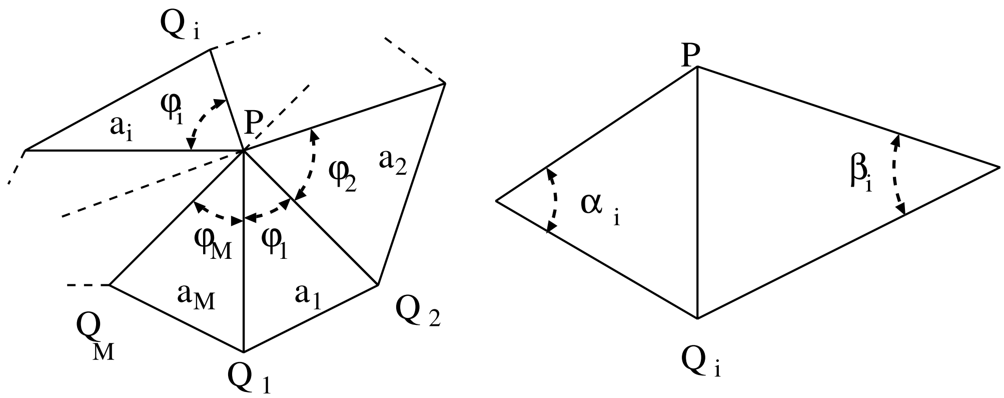


Figure 3.6: 1-Ring Neighborhood with Local Properties

Bi-Operator: For all operators, the squared operator can be defined by

$$\Lambda^2(\mathbf{P}) = \Lambda(\Lambda(\mathbf{P})), \tag{3.10}$$

where $\Lambda()$ is an operator and $\Lambda^2()$ is called a bi-*operator-name* operator. For example, a bi-umbrella operator is given by

$$L^2(\mathbf{P}) = L(L(\mathbf{P})) = \frac{1}{M}\sum_{i=1}^{M} L(\mathbf{Q}_i) - L(\mathbf{P}). \tag{3.11}$$

3.4 Visualization on Polygonal Surface

The Z-Score thresholding and two types of the color mapping methods are represented in this Section for the mean, Gaussian and principal curvatures visualization on triangulated surfaces. The curvature distributions are considered to estimate the polygonal surface fairness and smoothness [27]. The visualization methods extract roughly the ridge and ravine [2, 3, 4, 34, 42] which are the sets of the surface characteristic curves. The following methods are applied at only rendering triangles. This is a new method to visualize curvatures on the triangle meshes.

3.4.1 Z-Score Thresholding

The discrete approximation may include a fatal error, and the distribution of the discrete curvatures may include a very big deviation. If a set of estimated discrete curvature values has such kind of deviation, the appropriate normalization is difficult for a RGB color rendering. One possible approach is a Z-Score thresholding technique for discrete curvature values to eliminate such kind of fatal errors. The Z-Score of a value ϕ_i is calculated by

$$Z(\phi_i) = \frac{\phi_i - \hat{\phi}}{\sigma}, \quad \sigma = \sqrt{\frac{1}{M}\sum_{k=1}^{M}(\phi_i - \hat{\phi})^2}, \quad \hat{\phi} = \frac{1}{M}\sum_{i=1}^{M}\phi_i, \tag{3.12}$$

where $\hat{\phi}$ is the arithmetic mean of $\{\phi_i\}$ and σ is a deviation of the set $\{\phi_i : i = 1, 2, ...M\}$. The threshold function $Z_T()$ is given by

$$Z_T(\phi_i) = \begin{cases} \hat{\phi} - \sigma T \quad if \quad T \leq |Z(\phi_i)| \text{ and } \phi_i - \hat{\phi} < 0, \\ \hat{\phi} + \sigma T \quad if \quad T \leq |Z(\phi_i)| \text{ and } \phi_i - \hat{\phi} \geq 0, \\ \quad \phi_i \qquad\qquad\qquad otherwise, \end{cases} \tag{3.13}$$

where T is a threshold. The scalar functions on the surface can be visualized dynamically by changing this threshold T (In the numerical experiments, the initial T takes 1.5).

Geometrical meaning of the $Z_T(\text{curvature})$ is a low-pass filtered curvature which is a very roughly approximation of the Laplace-Beltrami operator of the curvature. For example, $Z_T(H)$ approximates $\triangle_B(H)$.

3.4.2 Color Mapping

Two useful color mapping functions are implemented in the thesis to recognize the surface feature, discrete smoothness and fairness for the human perception. One is a Pastel mapping to extract a distribution between extrema (maxima and minima) and average of the discrete curvatures. Another one is a Bipolar mapping to extract the extrema of the discrete curvatures. The following Figures 3.7, 3.8 and 3.9 represent the visualization of the implemented Pastel and Bipolar color mappings about the discrete mean curvature values.

Figure 3.7: Normal

Figure 3.8: Pastel

Figure 3.9: Bipolar

The general RGB color function domain is an interval $[0, 1]$ for each red, green and blue colors. The normalization function $Z_n()$ is given by

$$Z_n(\phi_i) = \frac{Z_T(\phi_i) - \mathbf{min}(Z_T(\phi_i))}{\mathbf{max}(Z_T(\phi_i)) - \mathbf{min}(Z_T(\phi_i))}, \tag{3.14}$$

where $\mathbf{min}()$ and $\mathbf{max}()$ return the minimum and maximum values in arguments, respectively.

Pastel: The Pastel map (see Figure 3.8 and 3.10) of RGB function is given by

$$RGB \equiv (R_{Pas}(Z_n(\phi_i)), G_{Pas}(Z_n(\phi_i)), B_{Pas}(Z_n(\phi_i))), \tag{3.15}$$

where a red $R_{Pas}()$, green $G_{Pas}()$ and blue $B_{Pas}()$ color ratio functions are defined by

$$R_{Pas}(\phi) = \begin{cases} 2\phi & if \quad 0 \leq \phi < \frac{1}{2}, \\ 1 & if \quad \frac{1}{2} \leq \phi \leq 1, \end{cases} \tag{3.16}$$

$$G_{Pas}(\phi) = \begin{cases} 1 & if \quad 0 \leq \phi < \frac{1}{2}, \\ -2\phi + 2 & if \quad \frac{1}{2} \leq \phi \leq 1, \end{cases} \tag{3.17}$$

$$B_{Pas}(\phi) = \begin{cases} -2\phi + 1 & if \quad 0 \leq \phi < \frac{1}{2}, \\ 2\phi - 1 & if \quad \frac{1}{2} \leq \phi \leq 1, \end{cases} \tag{3.18}$$

where ϕ takes an interval $[0, 1]$.

Bipolar: The Bipolar map (see Figure 3.9 and 3.11) of RGB function is given by

$$RGB \equiv (R_{Bip}(Z_n(\phi_i)), G_{Bip}(Z_n(\phi_i)), B_{Bip}(Z_n(\phi_i))), \tag{3.19}$$

where a red $R_{Bip}()$, green $G_{Bip}()$ and blue $B_{Bip}()$ color ratio functions are defined by

$$R_{Bip}(\phi) = \begin{cases} 0 & if \quad 0 \le \phi < \frac{1}{2}, \\ 2\phi - 1 & if \quad \frac{1}{2} \le \phi \le 1, \end{cases} \tag{3.20}$$

$$G_{Bip}(\phi) = -\sin\left(\arccos\left(2\phi - 1\right)\right) + 1, \tag{3.21}$$

$$B_{Bip}(\phi) = \begin{cases} -2\phi + 1 & if \quad 0 \le \phi < \frac{1}{2}, \\ 0 & if \quad \frac{1}{2} \le \phi \le 1, \end{cases} \tag{3.22}$$

where ϕ takes an interval $[0, 1]$.

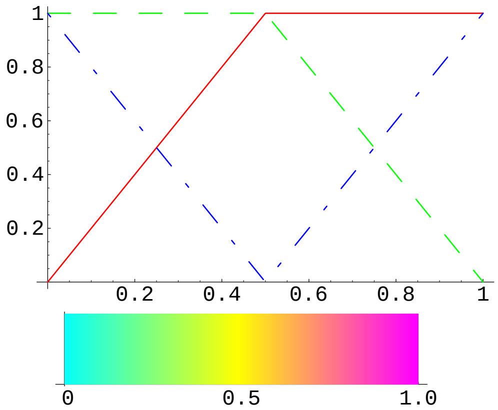


Figure 3.10: Pastel Mapping

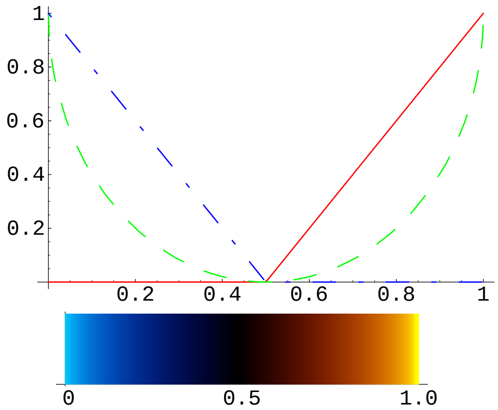


Figure 3.11: Bipolar Mapping

The following Figures from 3.12 to 3.17 represent the mean curvature visualization via Pastel (upper figures) and Bipolar (bottom figures) color mapping methods for different thresholds.

Figure 3.12: $T = 0.1$

Figure 3.14: $T = 1.5$

Figure 3.16: $T = 3.0$

Figure 3.13: $T = 0.1$

Figure 3.15: $T = 1.5$

Figure 3.17: $T = 3.0$

3.5 Curvature Approximations

The analytical surface mean and Gaussian curvatures are approximated by the six methods in this Section. Numerical experiments (see Section 3.5.5) show that the Laplace-Beltrami with the proposed new modified Desbrun method (**c**) (see Section 3.5.1) is the best method to approximate the mean and Gaussian curvatures for the irregular triangulated polygonal surface. Here $\approx$ represents an approximation.

Before representing the curvature approximation methods, a few implemented discrete surface properties are defined by the following paragraphs about the vertex $\mathbf{P}$.

Surface Normal: The discrete surface unit normal vector $\mathbf{N}$ is calculated by the following triangle area weighted cross products at an evaluation vertex $\mathbf{P}$.

$$\mathbf{N}_{f_i} = \overrightarrow{\mathbf{PQ}_i} \times \overrightarrow{\mathbf{PQ}_{i+1}}, \quad \mathbf{N}_{total} = \frac{1}{2}\sum_{i=1}^{M} \mathbf{N}_{f_i}, \quad \mathbf{N} = \frac{\mathbf{N}_{total}}{|\mathbf{N}_{total}|},$$

where $\mathbf{N}_{f_i}$ is a face normal vector of the triangle facet f_i, $\mathbf{N}_{total}$ is an approximated normal vector of the 1-ring neighborhood, $\times$ is a cross product, and $|\mathbf{V}|$ represents a size of the vector $\mathbf{V}$.

Surface Area: The discrete surface area is given by

$$a_i = \frac{1}{2}|\mathbf{N}_{f_i}|, \quad A = \sum_{i=1}^{M} a_i$$

where a_i is the triangle area and A is the 1-ring neighbor area.

Triangle Angle: The discrete triangle angle φ_i between $\overrightarrow{\mathbf{PQ}_i}$ and $\overrightarrow{\mathbf{PQ}_{i+1}}$ (see Figure 3.6) is calculated by the following equation.

$$\varphi_i = \arccos\left(\frac{\overrightarrow{\mathbf{PQ}_i} \cdot \overrightarrow{\mathbf{PQ}_{i+1}}}{|\overrightarrow{\mathbf{PQ}_i}||\overrightarrow{\mathbf{PQ}_{i+1}}|}\right), \tag{3.23}$$

where $\cdot$ is an inner product.

Directional Curvature: The discrete directional curvature k_i is approximated by

$$k_i \approx \frac{2\mathbf{N} \cdot (\mathbf{Q}_i - \mathbf{P})}{|\mathbf{Q}_i - \mathbf{P}|^2} \tag{3.24}$$

for an edge $\overrightarrow{\mathbf{PQ}_i}$ direction.

Projection Operator: The projection operator $\mathbf{V}_{proj\mathbf{N}^{\perp}}$ is defined by

$$\mathbf{V}_{proj\mathbf{N}^{\perp}} \equiv (\mathbf{I} - \mathbf{N}\mathbf{N}^t)\mathbf{V} = \mathbf{V} - (\mathbf{V} \cdot \mathbf{N})\mathbf{N}, \tag{3.25}$$

where I is a 3x3 identity matrix. The operator represents the vector which is projected $\mathbf{V}$ onto the plane defined by the normal vector $\mathbf{N}$.

3.5.1 Laplace-Beltrami with Gauss-Bonnet Methods

Three different curvature approximations based on the Laplace-Beltrami operator and Gauss-Bonnet theorem [8, 29] are described via the following popular formulas [5, 8, 33].

$$\triangle_B(\mathbf{S}) = 2H\mathbf{N}, \quad |K| = \lim_{\Delta S \to 0} \frac{\Delta A}{\Delta S} \tag{3.26}$$

$$\iint_\Omega K\, dA = 2\pi\chi(\mathbf{S}), \quad \int_{\partial\Omega} k_g\, ds + \iint_\Omega K\, dA = 2\pi - \sum_{i=1}^{M} \varphi_i, \tag{3.27}$$

where H and K represent the mean and Gaussian curvatures, respectively, $\chi(\mathbf{S})$ is the Euler characteristics of the surface $\mathbf{S}$, ΔA is the Gauss map area corresponding to the surface area ΔS, φ_i is the triangle angle defined in equation (3.23), and k_g is the geodesic curvature [8] of the boundary arc $\partial\Omega$. The left-hand and right-hand sides of the equation (3.27) represent the Gauss-Bonnet theorem for the closed and open surface, respectively. The methods can provide us with useful approximations of high-order (even number) derivatives of the mean and Gaussian curvatures. Because the Laplace-Beltrami operator can be applied to curvatures by the same precisions of curvature derivatives and curvatures. Instead of smooth continuous surface $\mathbf{S}$, $\mathbf{P}$ denotes the evaluation vertex of triangle meshes.

Desbrun Method (a)

The mean and Gaussian curvature were approximated by

$$H \approx \frac{1}{2}\mathbf{N} \cdot \triangle_{B_{Des}}(\mathbf{P}), \quad K \approx \frac{3}{A}(2\pi - \sum_{i=1}^{M} \varphi_i) \tag{3.28}$$

in [32]. The method is very good for the distribution of mean curvature and Gaussian curvature approximations (see Section 3.5.5) for the irregular meshes but the scale of the mean curvature and Gaussian curvature is a bad. Therefore this approximation can not be used to the methods which are constructed by the mean and Gaussian curvatures simultaneously (ex. principal curvatures).

Laplace-Beltrami Normal Method (b)

The Laplace-Beltrami operator of the surface unit normal is represented by $\triangle_B(\mathbf{N})$, and the inner product of this and the surface unit normal is given by

$$\triangle_B(\mathbf{N}) \cdot \mathbf{N} = -2(2H^2 - K), \tag{3.29}$$

from the equation (2.1). The mean and Gaussian curvatures approximated by one Laplace-Beltrami operator, where the Gaussian curvature is given by a solution of the simultaneous system of the equation

$$H \approx \frac{1}{2}\mathbf{N} \cdot \triangle_{B_{Des}}(\mathbf{P}), \quad K \approx \frac{1}{2}(\triangle_{B_{Des}}(\mathbf{N}) \cdot \mathbf{N}) + 2(\frac{1}{2}\mathbf{N} \cdot \triangle_{B_{Des}}(\mathbf{P}))^2. \tag{3.30}$$

This is a new method to approximate the Gaussian curvature. The advantage of the method is that the mean and Gaussian curvatures can be calculated by only one operator (Laplace-Beltrami operator). In the thesis, the Laplace-Beltrami operator of this method is approximated by the Desbrun's operator which was represented in equation (3.7). Therefore the irregular mesh tolerance of the Desbrun's operator is inherited to this method.

Derivation: *The equation (3.29) is derived by the following equations via the Laplace-Beltrami operator equation (2.1).*

$$\begin{aligned}\triangle_B(\mathbf{N})\cdot\mathbf{N} &= \frac{1}{\sqrt{EG-F^2}}\left(\frac{\partial}{\partial u}\frac{G\mathbf{N}_u - F\mathbf{N}_v}{\sqrt{EG-F^2}} + \frac{\partial}{\partial v}\frac{E\mathbf{N}_v - F\mathbf{N}_u}{\sqrt{EG-F^2}}\right)\cdot\mathbf{N} \\ &= \left(\frac{E\mathbf{N}_{vv} - 2F\mathbf{N}_{uv} + G\mathbf{N}_{uu}}{EG-F^2} + \frac{(\mathbf{N}_u f_1(u,v) + \mathbf{N}_v f_2(u,v))}{2(EG-F^2)^2}\right)\cdot\mathbf{N},\end{aligned}$$

where

$$\begin{aligned}f_1(u,v) &= E_vFG + 2FF_uG - 2EF_vG - E_uG^2 - 2F^2G_u + EGG_u + EFG_v, \\ f_2(u,v) &= -2E_vF^2 + 2EFF_v + EE_vG + E_uFG - 2EF_uG + EFG_u - E^2G_v.\end{aligned}$$

Now $\mathbf{N}_u\cdot\mathbf{N} = \mathbf{N}_v\cdot\mathbf{N} = 0$ *because of*

$$\frac{\partial}{\partial u}(\mathbf{N}^2 = 1) \Longrightarrow 2\mathbf{N}\cdot\mathbf{N}_u = 0, \quad \frac{\partial}{\partial v}(\mathbf{N}^2 = 1) \Longrightarrow 2\mathbf{N}\cdot\mathbf{N}_v = 0,$$

therefore,

$$\begin{aligned}\triangle_B(\mathbf{N})\cdot\mathbf{N} &= \left(\frac{E\mathbf{N}_{vv} - 2F\mathbf{N}_{uv} + G\mathbf{N}_{uu}}{EG-F^2}\right)\cdot\mathbf{N} \\ &= -\frac{E(-KG+2HN) - 2F(-KF+2HM) + G(-KE+2HL)}{EG-F^2} \\ &= -2\frac{H(EN-2FM+GL) + K(EG-F^2)}{EG-F^2} \\ &= -2(2H^2 - K),\end{aligned}$$

where the useful formulae

$$\mathbf{N}_{uu}\cdot\mathbf{N} = KE - 2HL, \quad \mathbf{N}_{uv}\cdot\mathbf{N} = KF - 2HM, \quad \mathbf{N}_{vv}\cdot\mathbf{N} = KG - 2HN,$$

are represented in [34] page 127.

Modified Desbrun Method (c)

The Desbrun's curvature approximation method **(a)** provides good mean and Gaussian curvature approximations for the irregular meshes, respectively. However the scale between estimated mean curvatures and Gaussian curvatures can take a very different scale, therefore these curvatures can not be applied to one method, simultaneously for example principal curvatures. The principal curvatures k_{max} and k_{min} are given by

$$k_{max} = H + \sqrt{H^2 - K}, \quad k_{min} = H - \sqrt{H^2 - K},$$

then approximated mean and Gaussian curvatures must satisfy an inequality $H^2 - K \geq 0$. If the scale between estimated curvatures is a very different scale, then the approximated mean and Gaussian curvatures may satisfy $H^2 - K < 0$.

New improved Desbrun method which is named by the modified Desbrun method is given by

$$H \approx \frac{1}{2}\mathbf{N} \cdot \triangle_{B_{Des}}(\mathbf{P}), \quad K \approx \lambda K_{Des}, \tag{3.31}$$

where

$$K_{Des} = \frac{3}{A}(2\pi - \sum_{i=1}^{M} \varphi_i), \tag{3.32}$$

$$\lambda = \min_{all\ points}\{\frac{H^2}{K_{Des}} : K_{Des} > H^2,\ H \neq 0\}. \tag{3.33}$$

The method solves the scale of the mean curvature and Gaussian curvature problem by scaling the Gaussian curvature to satisfy the condition $H^2 - K \geq 0$ without saddle point ($H = 0,\ K \neq 0$). This method is the best method to approximate the mean and Gaussian curvatures for the triangle meshes in six approximation methods from the numerical experiments in Section 3.5.5. This is a new method to approximate the Gaussian curvature of triangle meshes.

3.5.2 Taubin's Method (d)

The principal curvature approximation method was given by the curvature tensor analysis via the eigenvalues and eigenvectors of the certain 3x3 symmetric matrices defined by the matrix representation of curvature tensor in [12]. The Householder transformation and Givens rotation were applied to solve the eigenvalue problem (see Appendix B). The method provides us good results of the mean and Gaussian curvature approximations for the regular meshes. The method can be used to estimate the principal directions.

3.5.3 Central Difference Method (e)

The simple central difference with the local reparametrization method is developed in this Section. Let us consider two virtual grids on the tangent plane. One is defined by the arbitrary orthogonal vectors on the tangent plane. Another one is defined by the $\frac{\pi}{4}$ rotation of the first grid on the tangent plane then the eight intersection points can be calculated (see Figure 3.18) between the 1-ring neighborhood and two grids basis (four tangent lines).

In the numerical experiments, the tangents of the first grid is given by the second and third rows of the matrix $\mathbf{U}$ of equation (B.1) in Appendix B. The surface derivatives $\mathbf{S}_u, \mathbf{S}_v, \mathbf{S}_{uu}, \mathbf{S}_{uv}$, and $\mathbf{S}_{vv}$ are approximated by the solution of the simultaneous system of the following equations.

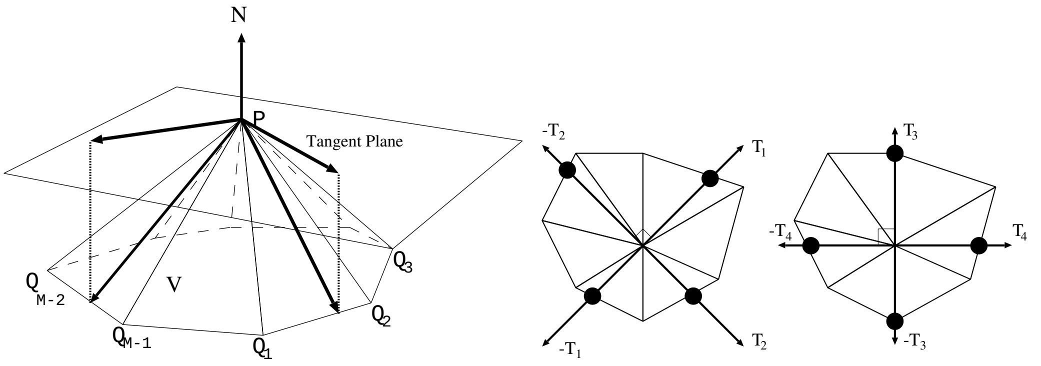


Figure 3.18: Virtual Grids

The surface is approximated by

$$\begin{aligned}\mathbf{S}(u+\alpha, v+\beta) \;&\approx\; \mathbf{S}+\alpha\mathbf{S}_u+\beta\mathbf{S}_v+\alpha\beta\mathbf{S}_{uv}+ \\ &+\frac{1}{2}(\alpha^2\mathbf{S}_{uu}+\beta^2\mathbf{S}_{vv}+\alpha^2\beta\mathbf{S}_{uuv}+\alpha\beta^2\mathbf{S}_{uvv})+\frac{1}{4}(\alpha^2\beta^2\mathbf{S}_{uuvv}),\end{aligned} \tag{3.34}$$

where α and β are constants via the Taylor series expansion. The eight intersection points are given by

$$\begin{aligned}&\mathbf{S}(u-d_{u_1}, v), \mathbf{S}(u+d_{u_2}, v), \mathbf{S}(u, v-d_{v_1}), \mathbf{S}(u, v+d_{v_2}), \\ &\mathbf{S}(u+d_{u_3}, v+d_{v_3}), \mathbf{S}(u+d_{u_3}, v-d_{v_4}), \mathbf{S}(u-d_{u_4}, v+d_{v_3}), \mathbf{S}(u-d_{u_4}, v-d_{v_4}),\end{aligned} \tag{3.35}$$

where $u_{1,2,3,4}$ and $v_{1,2,3,4}$ are the distances between $\mathbf{P}$ and intersection points.

3.5.4 Trapezoid Method (f)

The classical Euler formula about the normal curvature $k_n(\theta)$ is as follows:

$$k_n(\theta) = k_{max}\cos^2\theta + k_{min}\sin^2\theta, \tag{3.36}$$

where θ is the rotation parameter of the normal section plane. The total integral of the normal curvature and squared normal curvature [19] are given by

$$\frac{1}{2\pi}\int_0^{2\pi} k_n(\theta)\, d\theta = H, \quad \frac{1}{2\pi}\int_0^{2\pi} k_n^2(\theta)\, d\theta = \frac{1}{2}(3H^2-K). \tag{3.37}$$

The trapezoid approximation of the equation (3.37) is given by

$$\int_0^{2\pi} k_n(\theta)\, d\theta \approx \sum_{i=1}^{M}\varphi_i(\frac{k_i+k_{i+1}}{2}), \tag{3.38}$$

$$\int_0^{2\pi} k_n^2(\theta)\, d\theta \approx \sum_{i=1}^{M}\varphi_i(\frac{k_i^2+k_{i+1}^2}{2}), \tag{3.39}$$

where k_i is a discrete directional curvature and φ_i is the discrete triangle angle (see Figure 3.6). Therefore, the mean and Gaussian curvatures are approximated by a solution of the simultaneous system of the above equations (3.37), (3.38) and (3.39). This method was proposed in [23] for the discrete curvature extrema detection method.

3.5.5 Numerical Experiments

The following Figures represent the numerical experiments of the represented curvature approximation methods via the Bipolar color mapping where the threshold T takes 0.1. The vertex distribution of original 1-handle is separated into upper and lower parts by the horizontal center of Figure 3.19. The vertex distribution of original 2-handle is characterized by the regular subdivision in Figure 3.20. The quality of approximation methods can be evaluated by the observations of the curvature distribution discontinuous and sharp (high curvature) parts.

Figure 3.21 represents mean curvature approximation via methods **(a)**, **(b)** and **(c)**. Their curvature distribution is a good distribution around a boundary of vertex density.

Figures 3.23 and 3.24 represent that the polygonal surfaces have the mean curvature discontinuities at the boundaries of vertex distributions, therefore the Taubin's method **(d)** and trapezoid method **(f)** are the good approximation methods for regular meshes, but these methods provide the bad mean curvature approximation results for the irregular vertex distribution parts.

Figures 3.27, 3.29, 3.30, 3.33, 3.35 and 3.36 represent that the polygonal surfaces have the Gaussian curvature and minimum principal curvature discontinuities at the boundaries of vertex distributions, therefore the Taubin's method **(d)**, central difference method **(e)** and trapezoid method **(f)** provide the bad Gaussian curvature and minimum principal curvature approximation results for us, where the polygonal surface has the irregular vertex distribution part.

Figure 3.31 has two black circles. This means that the minimum principal curvature approximation is a wrong result, and the scale of the mean and Gaussian curvatures is a bad scale via the Desbrun method **(a)**.

Figures 3.37 and 3.38 represent that the Desbrun method **(a)** and the modified Desbrun method **(c)** are very good approximations of the distribution of the Gaussian curvature because the corners of 2-handle are extracted by the methods.

The modified Desbrun method **(c)** is the best approximation method in represented six methods for mean and Gaussian curvatures and its scale for the irregular triangle meshes from the numerical results.

The following Table 3.1 represents notation, name and equation number of the methods.

Notation	Name	Equation No.	Section
Method **(a)**	Desbrun Method	3.28	3.5.1
Method **(b)**	Laplace-Beltrami Normal Method	3.29, 3.30	3.5.1
Method **(c)**	Modified Desbrun Method [Best]	3.31, 3.32, 3.33	3.5.1
Method **(d)**	Taubin's Method	Appendix B	3.5.2
Method **(e)**	Central Difference Method	3.34, 3.35	3.5.3
Method **(f)**	Trapezoid Method	3.37, 3.38, 3.39	3.5.4

Table 3.1: Curvature Approximation Methods

Irregular Handles

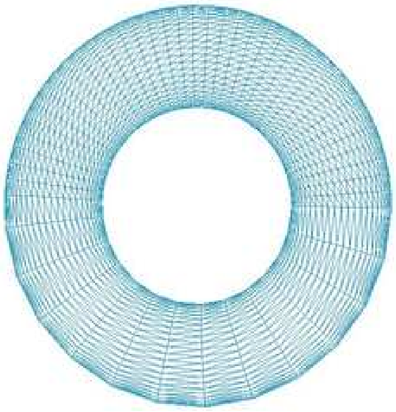


Figure 3.19: Meshes 1

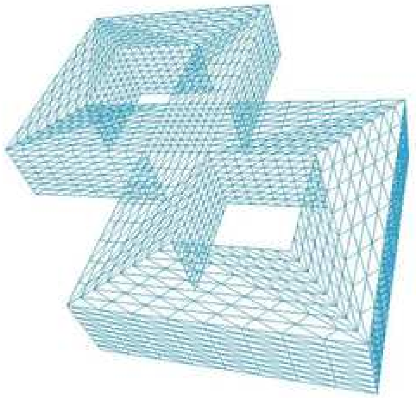


Figure 3.20: Meshes 2

Mean Curvature Bipolar Mapping

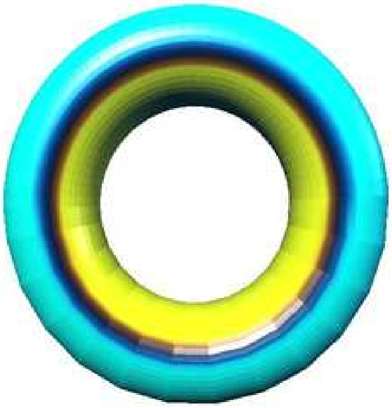


Figure 3.21: Method **(a)**,**(b)** and **(c)**

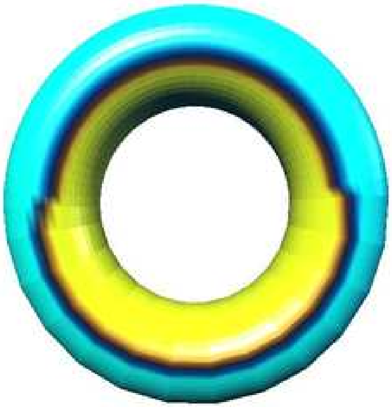


Figure 3.23: Method **(d)**

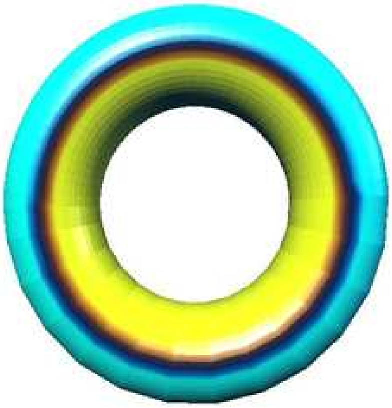


Figure 3.22: Method **(e)**

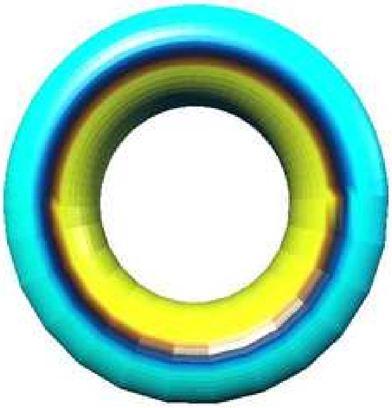


Figure 3.24: Method **(f)**

Gaussian Curvature Bipolar Mapping

Figure 3.25: Method **(a)**

Figure 3.28: Method **(b)**

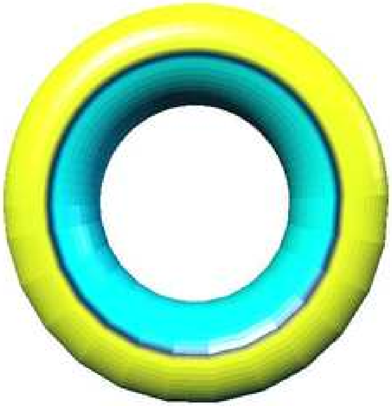


Figure 3.26: Method **(c)**

Figure 3.29: Method **(d)**

Figure 3.27: Method **(e)**

Figure 3.30: Method **(f)**

k_{min} **Bipolar Mapping**

Figure 3.31: Method **(a)**

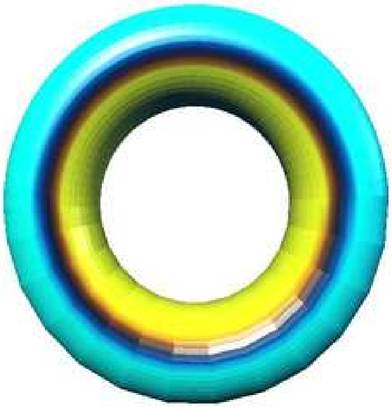


Figure 3.34: Method **(b)**

Figure 3.32: Method **(c)**

Figure 3.35: Method **(d)**

Figure 3.33: Method **(e)**

Figure 3.36: Method **(f)**

Gaussian Curvature Bipolar Mapping

Figure 3.37: Method **(a)**

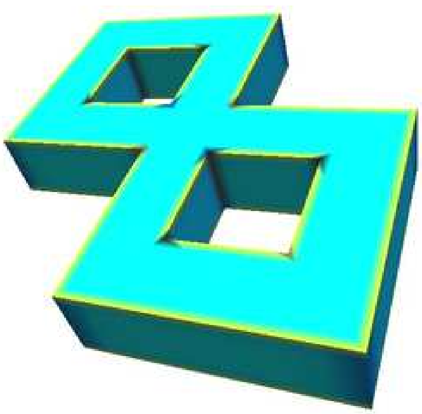


Figure 3.40: Method **(b)**

Figure 3.38: Method **(c)**

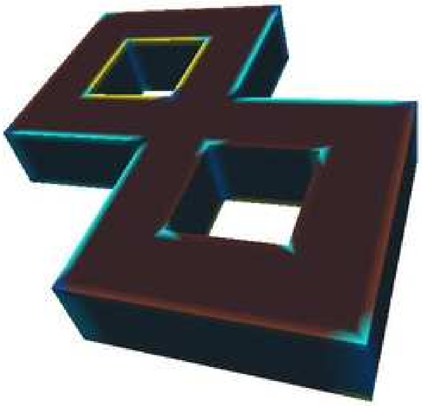


Figure 3.41: Method **(d)**

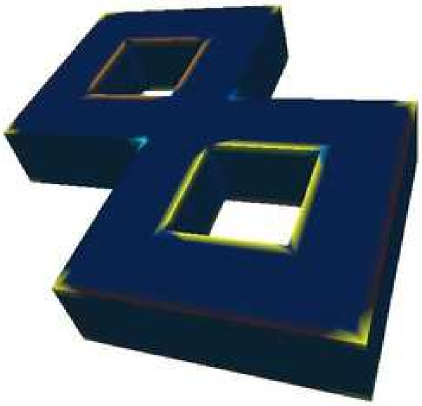


Figure 3.39: Method **(e)**

Figure 3.42: Method **(f)**

3.6 Topological Methods: Multi-Grid Approach

The simple topological methods are introduced in this Section. The edge collapse [15] technique is used for the mesh simplification (decimation). The regular subdivision [15] technique is used for the polygonal interpolation. The simultaneous subdivision and decimation method is called a multi-grid approach [37] which have many excellent applications (the flow acceleration, polygonal data compression, etc).

Regular Subdivision: The triangle is subdivide to four triangles (see Figure 3.43). Inserted vertex is located on a center of the edge.

Simple Decimation (Edge Collapse): The edge is collapsed by the given priority queue (see Figure 3.43).

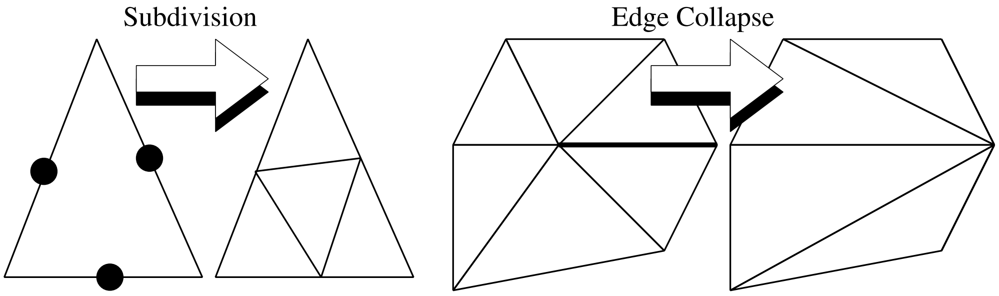


Figure 3.43: Triangle Subdivision and Edge Collapse

3.7 Singularities Developed by Discrete Surface Flows

The polygonal singularity is defined in this Section instead of the smooth continuous surface singularity of Section 2.3. The equation (2.28) means that a singular vertex has the zero area because of the relation $EG - F^2 = A^2$. Therefore, the evaluation vertex $\mathbf{P}$ is a polygonal surface singularity vertex where the 1-ring neighbor area A is less than a computer-ϵ:

$$A = \sum_{i}^{M} a_i \leq \text{computer-}\epsilon << 1.$$

In the numerical experiments, computer-ϵ takes a minimum absolute value of the double floating-point value.

The polygonal singularities can be developed in case of no smooth continuous surface singularities. Because the discrete normal vector flow directions converge to their skeleton of the polygonal surface (Voronoi sites) [3, 4, 14, 28, 42]. For example, the Desbrun's discrete mean curvature flow [31] can be developed the polygonal singularities because of their surface normal flow directions [41].

Chapter 4

Applications

The chapter is devoted to three triangulated polygonal modeling applications which are the surface smoothing, mesh regularization, and fair surface modeling based on the considered variational surface calculus in Chapter 2 via the developed discrete differential geometry in Chapter 3. New surface flows are proposed for modeling these applications. The surface smoothing is applied by the discrete mean curvature flows to minimize the membrane energy on the given triangle meshes. The mesh regularization is achieved by the tangent flows which are constructed by the projection vectors onto the tangent plane to uniformize vertices. The fair surface is modeled by the discrete approximation of the elastic flows. The implemented mean curvature flows are four types of the discrete approximations in Section 4.1. Area weighted Laplace-Beltrami flow (MC-Flow **(d)**) is the best method for the triangulated polygonal surface smoothing about the flow stability and vertex distribution independency criterions from the numerical experiments. The modeled tangent flows are six types of the discrete implementations of vertex diffusions on the surface in Section 4.2. Area weighted bi-Laplace-Beltrami tangent flow (T-Flow **(f)**) provides us the best results because of their high quantities of shape preservation and vertex uniformization. The developed elastic flows are five types of the discrete approximations in Section 4.3. Elastica flow (Flow **(e)**) has the best quantity of the fairness and smoothness from numerical experiments.

4.1 Surface Smoothing via Mean Curvature Flows

The surface smoothing is a very useful not only for computer graphics but also for an arbitrary data noise reduction processing because the smoothing effect is dominated by the physical model (ex. heat equation). The considered approximations of the mean curvature flow are the Laplacian flow [11, 13, 31], mean curvature flow [31], modified mean curvature flow [48] and area weighted Laplace-Beltrami flow [41].

4.1.1 Laplacian Flow (a)

The Laplacian flow is approximated by

$$\mathbf{P}_{new} = \mathbf{P}_{old} + \epsilon L(\mathbf{P}_{old}), \tag{4.1}$$

where ϵ must satisfy an inequality $0 < \epsilon \leq \frac{1}{2}$ because this is a linear differential equation. This flow has the excellent stability and high evolution speed but the flow strongly depends on the vertex distribution of triangle meshes for the same geometry part [41].

Taubin analyzed the details of the Laplacian flow, low-pass filtering and its numerical implementations in $[11,\ 13]$ about the surface smoothing purposes.

4.1.2 Mean Curvature Flow (b)

The mean curvature flow is approximated by the represented (see equation (3.7)) Desbrun's Laplace-Beltrami operator [31]. The evolution equation is given by

$$\mathbf{P}_{new} = \mathbf{P}_{old} + \epsilon \frac{1}{2} \triangle_{B_{Des}}(\mathbf{P}_{old}), \tag{4.2}$$

where ϵ takes $\epsilon = \frac{\mathbf{min}(A)}{2}$ [41] from consideration of the numerical analysis of the heat equation problem, where $\mathbf{min}(A)$ returns a minimum area of the 1-ring neighborhood. This flow provides us a good approximation of the mean curvature flow but the polygonal singularities may be developed by this flow [41].

4.1.3 Ohtake's Mean Curvature Flow (c)

Ohtake et al. developed a variant of the mean curvature flow in order to avoid polygonal singularities [48] via adding the tangent component to the speed vector whose evolution equation is defined by

$$\mathbf{P}_{new} = \mathbf{P}_{old} + \epsilon \mathbf{F}, \tag{4.3}$$

where $\mathbf{F}$ is given

$$\mathbf{F} = \begin{cases} |H|\mathbf{L}_n \cos\theta, & if \quad \cos\theta > \alpha, \\ 2H\mathbf{N} - |H|\mathbf{L}_n \cos\theta, & if \quad \cos\theta < -\alpha, \\ \mathbf{0}, & if \quad |\cos\theta| \leq \alpha, \end{cases} \tag{4.4}$$

where $\mathbf{L}_n$ represents the normalized umbrella operator, θ is the angle between the mean curvature normal and umbrella operator, and α is the experimental value ($\alpha = 0.1$).

4.1.4 Area Weighted Laplace-Beltrami Flow (d)

The area weighted Laplace-Beltrami flow [41] is defined by following discrete evolution equation:

$$\mathbf{P}_{new} = \mathbf{P}_{old} + \epsilon \frac{1}{2} \triangle_{B_{Area}}(\mathbf{P}_{old}). \tag{4.5}$$

This flow has the interim approximation effects between the Laplacian flow and mean curvature flow, therefore this flow achieves the stable flow and vertex distribution independency.

4.1.5 Comparison of Methods

Figure 4.1: MC-Flow **(a)** 100 iterations

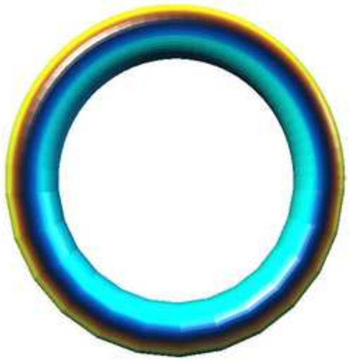


Figure 4.4: MC-Flow **(b)** 250 iterations

Figure 4.2: MC-Flow **(c)** 2000 iterations

Figure 4.5: MC-Flow **(d)** 1500 iterations

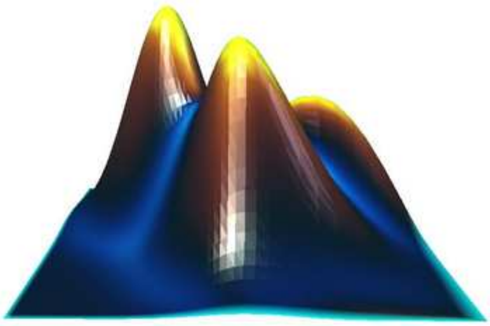


Figure 4.3: MC-Flow **(a)** 30 iterations

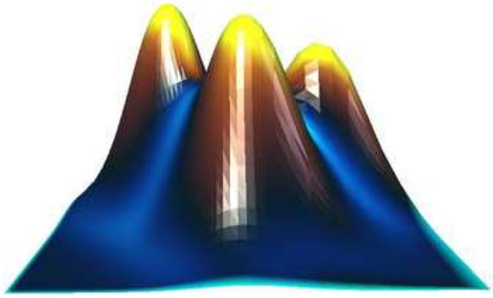


Figure 4.6: MC-Flow **(d)** 200 iterations

The represented smoothing methods are evaluated by the above two types of the irregular meshes, where MC-Flow **(a)**, **(b)**, **(c)** and **(d)** are defined in the following Table 4.1.

Notation	Flow Name	Equation No.	Section
MC-Flow **(a)**	Laplacian	3.3, 4.1	4.1.1
MC-Flow **(b)**	Desbrun's Mean Curvature	3.7, 4.2	4.1.2
MC-Flow **(c)**	Ohtake's Mean Curvature	4.3, 4.4	4.1.3
MC-Flow **(d)**	Area Weighted Laplace-Beltrami [Best]	3.9, 4.5	4.1.4

Table 4.1: Mean Curvature Flow Approximations

The original torus (tri-mountain) is meshed by the irregular vertex distribution separated into the up (left) and down (right) parts, see Figure 4.7.

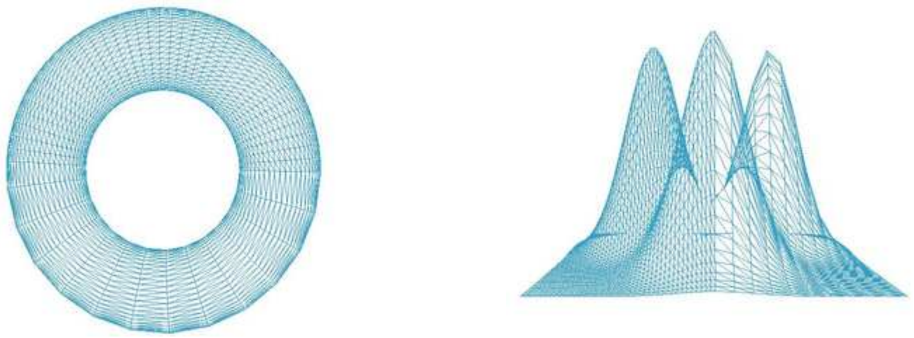
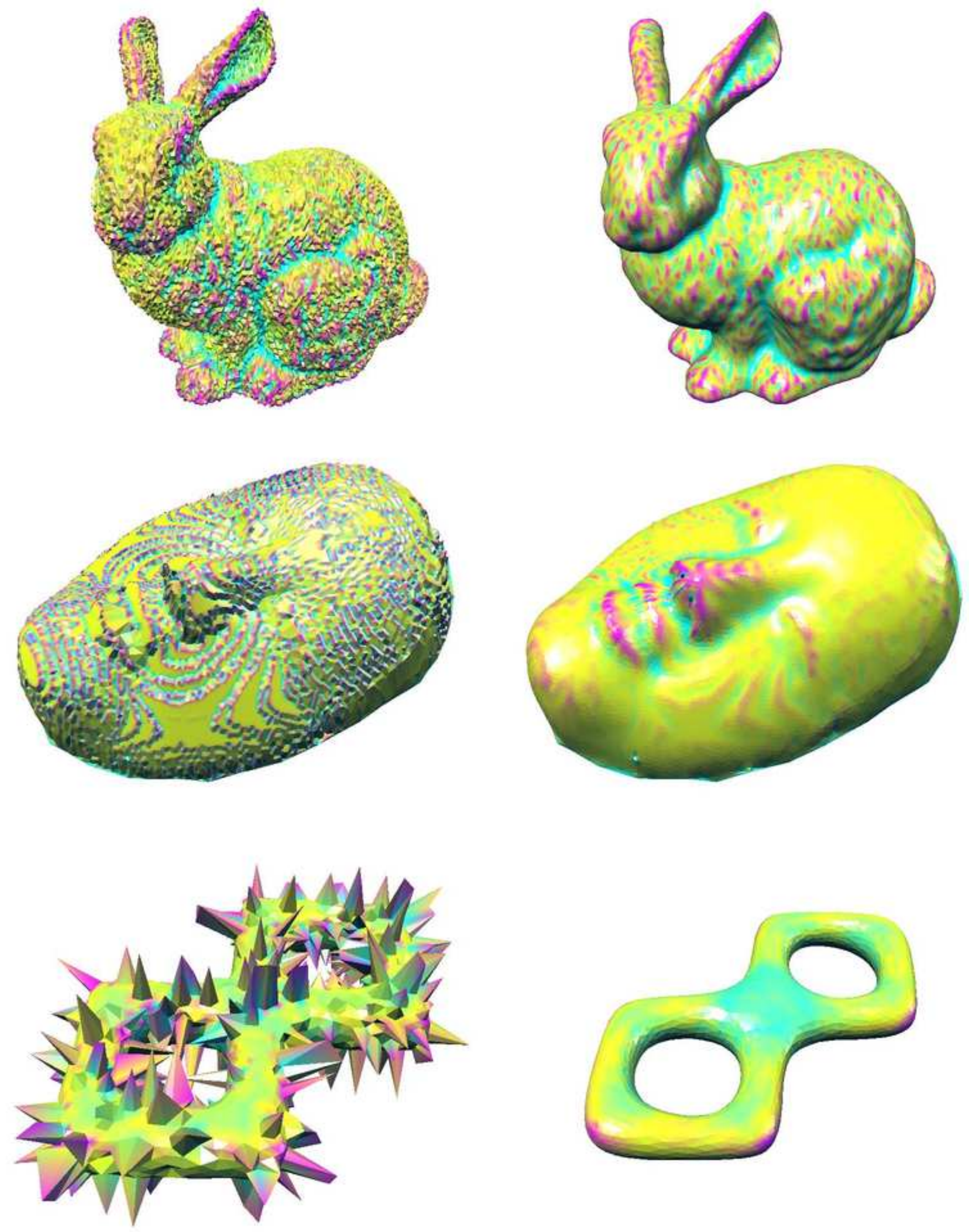


Figure 4.7: Original Meshes

The mean curvature flow MC-Flow **(b)** and modified mean curvature flow MC-Flow **(c)** develop the polygonal singularities at tops of the tri-mountain, therefore the evolution may be stopped before removing noises. It means that these flows always does not work correctly to arbitrary triangle meshes.

Figures 4.1 and 4.3 show that the Laplacian flow MC-Flow **(a)** strongly depends on the vertex distribution for the same geometry. Figure 4.4 represent the Desbrun's Laplace-Beltrami operator is a very good approximation of the mean curvature normal vector, because the evolving torus does not depend on the distribution of the vertices for the same geometry. The modified mean curvature flow can be develop the mean curvature discontinuities, see the center circle (mean curvature equal zero circle) of the torus in Figure 4.2. Figures 4.5 and 4.6 show that the area weighted Laplace-Beltrami flow achieves the good flow to avoid the such kind of the mean curvature discontinuities and vertex distribution dependency.

Smoothing Examples: The following Figures 4.8 represent the smoothing results via the proposed new flow MC-Flow **(d)** which show that the method is a very useful for noise reductions of the complex triangulated polygonal surfaces which have the non-uniform vertex distributions with a few noise types. The result of First-Bunny smoothing represents that the flow MC-Flow **(d)** has a high persistence for the large polygon data and uniform small noise distribution. The smoothing results of Third-2-Handles and Fourth-Bottle guarantee the high stability of the flow MC-Flow **(d)** because of their complex noisy sharp parts. Particularly, the smoothing result of the Second-Noh-Mask is very important in the shape modeling and analysis, because this Noh-Mask is a real-world data with a real-world noise which is produced by a laser scanner.



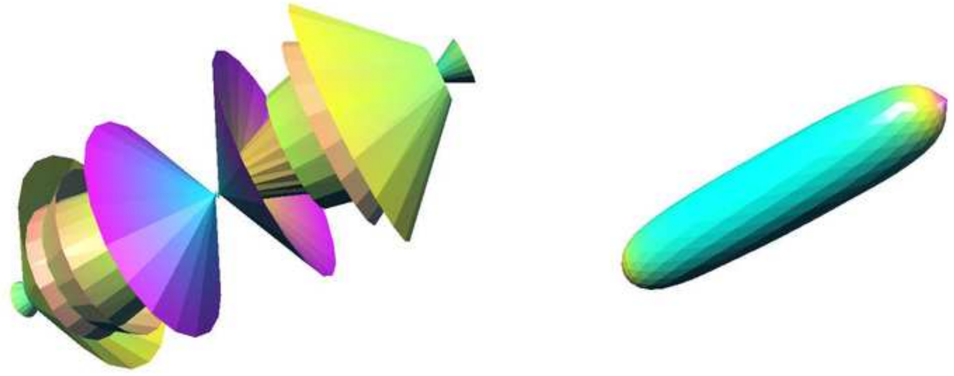


Figure 4.8: Complex Shape Smoothing:

Left figures are the extreme noisy surfaces. Right figures represent denoising surfaces by the area weighted Laplace-Beltrami flow (see Section 4.1.4). First-Bunny has the uniform small noises, the uniform meshes and a few holes. Second-Noh-Mask is a real-world data (produced by a laser scanner), and has the irregular meshes and a few holes. Third-2-Handles has a few self-intersections, the semi-irregular meshes, and the non-uniform sharp noises. Fourth-Bottle has the uniform meshes, a few steep parts and self-intersections.

4.2 Reparametrization via Tangent Flows

When we design the surface by triangulated meshes, the desirable mesh is the regular triangle and uniform vertex distribution in many cases because of many useful reasons (ex. surface rendering). One possible approach is the restricted vertex diffusion onto the tangent plane which is named a tangent flow. The thesis proposes a new discrete tangent flow, and shows the comparison of new methods with old methods. Examples of the vertex uniformization are given by the following Figure 4.10. The reparameterized Venus (see Figure 4.10) is constructed by more uniform vertex distribution and regular triangles than an original Venus (see Figure 4.9) via the tangent flow.

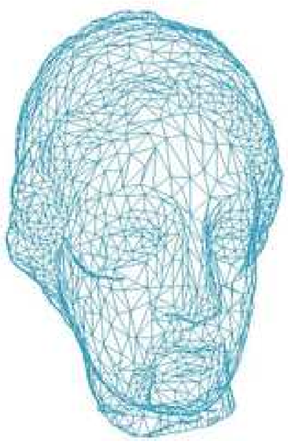


Figure 4.9: Original Venus Meah

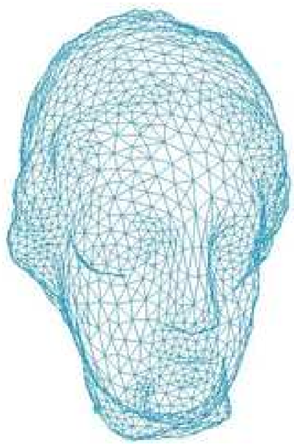


Figure 4.10: Reparameterized Venus

The analytic tangent flow is described by the following evolution equation.

$$\mathbf{S}_t = F_1\mathbf{T}_1 + F_2\mathbf{T}_2, \quad \mathbf{T}_1 \perp \mathbf{T}_2, \tag{4.6}$$

where $\mathbf{S}_t$ is a time derivative of the surface with respect to the time parameter t, F_1 and F_2 are the scalar functions, and $\mathbf{T}_1$ and $\mathbf{T}_2$ are the unit tangent vectors. The surface is modified along their tangents. It means that the analytic surface does not change by the tangent flow. The analytical tangent flow is equivalent to the surface reparametrization.

An arbitrary plane vector can be represented by two orthogonal vectors on its plane. Then, the equation (4.6) is represented by

$$\mathbf{S}_t = F\mathbf{T}, \quad \mathbf{T} \perp \mathbf{N}, \tag{4.7}$$

where $\mathbf{T}$ is an unit tangent vector and $\mathbf{N}$ is an unit normal vector.

The above equation (4.7) is discretized by

$$\mathbf{S}_{new} \longleftarrow \mathbf{S}_{old} + \epsilon F_{old}\mathbf{T}_{old}, \quad \epsilon << 1,$$

via the forward Euler difference method for the time parameter domain. Each vertices are updated by their speed vectors $F\mathbf{T}$ to new positions.

The types of treated tangent flows achieve the vertex uniformization. These tangent flows have one interesting property which is the stability influence for general surface flows whose vertices are displaced by their normal directions. Because these flow can develop the polygonal singularities on the skeletal sheets [3, 42]. If tangent flow provides us not only regular triangles but also more uniform distribution of the vertices, the high stability of evolution is expected via the tangent flow. Therefore the uniform distribution of the vertices are very important for the numerical implementation of the normal direction surface flows.

4.2.1 Speed Vector Modeling for Desirable Tangent Flow

Let us summarize the desirable effectiveness and conditions of the speed vector approximation for the tangent flow. Speed vector

- does not change the surface geometry (preserve feature curves, topology).
- diffuses the vertices (uniform vertex distribution on the surface).
- achieves the regular triangle meshes.

The following reparameterization operations are considered in the thesis via vertex diffusions onto the surface for the following 1-ring neighborhood Figure 4.11.

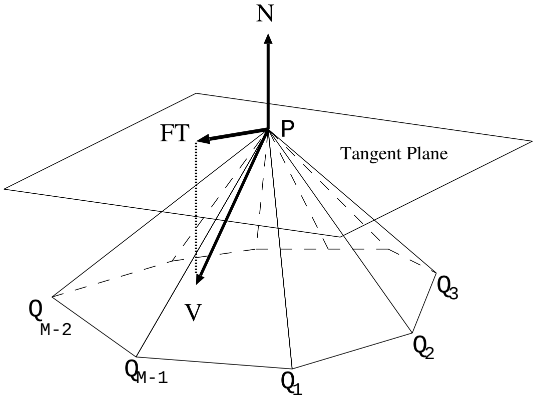


Figure 4.11: Tangent Plane with 1-Ring Neighborhood

The discrete approximations of the normal vector and projection operator were represented in Section 3.5.

4.2.2 Laplacian on $\mathrm{N}^{\perp}$: T-Flow (a)

The Laplacian operator diffuses the vertices, and smoothing the surface. For the tangent flow, speed vector is approximated by the projection of the umbrella operator (see equation (3.3) in Section 3.3) of the evaluation vertex $\mathbf{P}$ onto the tangent plane:

$$F\mathbf{T} = L(\mathbf{P})_{proj\mathbf{N}^{\perp}}. \tag{4.8}$$

The ϵ must be satisfy the inequality $0 < \epsilon \leq \frac{1}{2}$ for the stable flow. It was first proposed in [48].

4.2.3 Bi-Laplacian on $\mathbf{N}^{\perp}$: T-Flow (b)

The bi-Laplacian operator minimizes the thin plate energy on the flat metric (see Section 2.1.2) which means the surface shape is deformed more smoothly than the Laplacian operator. For the tangent flow, the speed vector is approximated by the projection of the bi-umbrella operator onto the tangent plane:

$$F\mathbf{T} = (-L^2(\mathbf{P}))_{proj\mathbf{N}^{\perp}}. \tag{4.9}$$

The ϵ must be satisfy the inequality $0 < \epsilon \leq \frac{1}{2}$ for the stable flow. It was proposed and used in [51].

4.2.4 Brakke's Vertex Averaging: T-Flow (c)

In [22], the vertex averaging method was proposed in 1992. This method preserves the 1-ring neighbor cone volume via the area weighted average of the centroids of the 1-ring neighbor triangles which can be formulated by

$$\mathbf{V}_{ave} = \frac{1}{3A}\sum_{i=1}^{M}\mathbf{N}_{f_i}\cdot(\mathbf{P}+\mathbf{Q}_i+\mathbf{Q}_{i+1}), \tag{4.10}$$

$$\lambda = \frac{1}{A^2}(\mathbf{V}_{ave}-\mathbf{P})\cdot\mathbf{N}_{total}, \tag{4.11}$$

where $\mathbf{V}_{ave}$ is a centroid. The speed vector is given by

$$F\mathbf{T} = \mathbf{V}_{ave} - \lambda\mathbf{N}_{total} - \mathbf{P}, \tag{4.12}$$

where the last equation (4.12) is modified for the tangent flow comparing for the original method in [22]. Because this method is denoted for the tangent flow. The ϵ can take 1 for the stable flow.

4.2.5 Ohtake's threshold approach on $\mathbf{N}^{\perp}$: T-Flow (d)

The combination smoothing with mesh regularization method was described in [48] via the Desbrun's Laplace-Beltrami operator and the umbrella operator projection onto the tangent plane. The speed vector of the tangent flow is as follows:

$$F\mathbf{T} = \mathbf{F}_{proj\mathbf{N}^{\perp}}, \tag{4.13}$$

where the vector $\mathbf{F}$ is given by the equation (4.4) in Section 4.1.3. The ϵ must be satisfy the inequality $0 < \epsilon \leq \frac{\mathbf{min}(A)}{2}$ for the stability purposes.

4.2.6 Mean Curvature Normal on $\mathbf{N}^{\perp}$: T-Flow (e)

For the strongly irregular meshes, the Desbrun's Laplace-Beltrami operator [32] (see equation (3.8) in Section 3.3) is very useful for the mean curvature normal approximation (see Section 3.5) which has a high accuracy than the area weighted Laplace-Beltrami operator for the normal vector approximation. The difference between the area weighted normal and Desbrun's operator can be extracted by

$$F\mathbf{T} = \triangle_{B_{Des}}(\mathbf{P})_{proj\mathbf{N}^{\perp}}. \tag{4.14}$$

4.2.7 Area Weighted Bi-Laplace-Beltrami Flow on $\mathrm{N}^{\perp}$: T-Flow (f)

Another possible approximation of the Laplace-Beltrami operator is the area weighted normalized umbrella operator (see equation (3.9) in Section 3.3) which was developed for the stable discrete mean curvature flow of the irregular meshes [41]. This operator has the middle effects between the umbrella operator and the Desbrun's Laplace-Beltrami operator about the vertex diffusion and the shape smoothing. The area weighted bi-Laplace-Beltrami tangent flow is constructed by

$$F\mathbf{T} = (-\triangle^2_{B_{Area}}(\mathbf{P}))_{proj\mathbf{N}^{\perp}}. \tag{4.15}$$

The ϵ must be satisfy the inequality $0 < \epsilon \leq 48\mathbf{min}(A)$ for the stable flow. For the acceleration and stability, the first few iterations are applied the tangent flow by the bi-umbrella tangent flow represented in equation (4.9). If $\mathbf{Q}_k$ is the boundary vertex, $\triangle_{B_{Area}}(\mathbf{Q}_k)$ takes zero. This is a new method to regularize the triangle meshes.

4.2.8 At Boundary Vertex

For the boundary vertices, the tangent vector is approximated via the central difference approximation of the boundary closed curve. It means that the boundary vertex is moved along the boundary closed curve. The tangent vector of the boundary closed curve $\mathbf{T}_{boundary}$ is given by central difference method:

$$\mathbf{T}_{boundary} \;=\; \frac{\mathbf{Q}_{B_1} - \mathbf{P}}{d_1} + \frac{\mathbf{P} - \mathbf{Q}_{B_2}}{d_2} - \frac{\mathbf{Q}_{B_1} - \mathbf{Q}_{B_2}}{d_1 + d_2},$$

where $\mathbf{Q}_{B_1}$ and $\mathbf{Q}_{B_2}$ are the successive boundary vertices, and d_i is a distance between the evaluation vertex $\mathbf{P}$ and $\mathbf{Q}_{B_i}$.

The flow vector is approximated by the weighted bi-Laplacian (squared umbrella operator) for the boundary closed curve. The weighted bi-Laplacian of the closed curve is given by

$$L_{curve}(\mathbf{P}) = \frac{1}{2}(L(\mathbf{Q}_{B_1}) - 2L(\mathbf{P}) + L(\mathbf{Q}_{B_2})), \tag{4.16}$$

where $L(\mathbf{V})$ is the umbrella operator in equation (3.3). It means that the operator $L_{curve}(\mathbf{P})$ is constructed by the two types of the umbrella operators which are the standard umbrella operator for the polygonal surface and the weighted umbrella operator for the closed curve. The speed vector is given by

$$F\mathbf{T} = (((-L_{curve}(\mathbf{P})) \cdot \mathbf{T}_{boundary}) \cdot \mathbf{T}_{boundary})_{proj\mathbf{N}^{\perp}}, \tag{4.17}$$

where $\mathbf{N}$ is the unit normal vector of the boundary vertex which is approximated by the area weighted cross products (same method of inner vertex case). This is a new method to regularize the closed boundaries.

4.2.9 Comparison of Methods

The surface reparameterization methods are evaluated by the non-uniform Gaussian-kernel (same $\epsilon = 0.1$), see Figure 4.12, where T-Flow **(a)**, **(b)**, **(c)**, **(d)**, **(e)** and **(f)** correspond to the following Table 4.2.

Notation	Method Name	Equation No.	Section
T-Flow **(a)**	Laplacian Tangent Flow	3.3, 4.8	4.2.2
T-Flow **(b)**	Bi-Laplacian Tangent Flow	3.11, 4.9	4.2.3
T-Flow **(c)**	Brakke's Vertex Averaging	4.10, 4.11, 4.12	4.2.4
T-Flow **(d)**	Ohtake's Threshold Approach	4.4, 4.3, 4.13	4.2.5
T-Flow **(e)**	Mean Curvature Tangent Flow	3.7, 4.14	4.2.6
T-Flow **(d)**	Area Weighted $\triangle_B$ Tangent Flow [Best]	3.9, 4.15	4.2.7

Table 4.2: Tangent Flow Implementations

The tangent flow methods T-Flow **(a)**, **(c)** and **(d)** destroy remarkablely the top part of the original shape, see Figure 4.13, 4.15, 4.16, 4.18, 4.19 and 4.22.

The Figures 4.20 and 4.23 show that the tangent flow method T-Flow **(e)** does not work as the mesh regularization expected.

The tangent flow method T-Flow **(b)** bends a steep part of the original Gaussian-kernel, see Figures 4.14 and 4.17.

The drift phenomenon which is a vertex shift phenomenon depending on geometry and its vertex density is developed by the Laplacian tangent flow T-Flow **(a)**, see Figures from 4.25 to 4.36. The bi-Laplacian tangent flow T-Flow **(b)** is inherited this drift phenomenon from the umbrella operator.

The reparameterized Figures 4.21 and 4.24 show that the area weighted bi-Laplace-Beltrami tangent flow T-Flow **(f)** is the best method for vertex uniformization with triangle regularization of the irregular triangle meshes, because the evolving surfaces were remeshed to satisfy almost the geometry preservation (particularly sharp parts), regular triangle meshing and uniform distribution of the vertices on the polygonal surface. The flow speed is low, but the proposed new method has important properties which are the drift phenomenon independency and the high sharp part dependency.

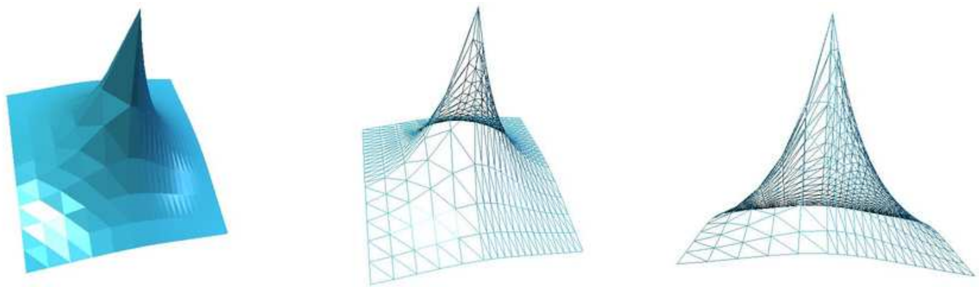


Figure 4.12: Original Model

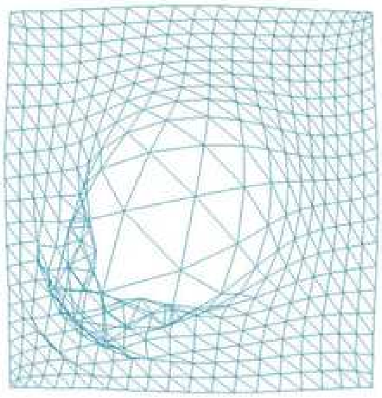


Figure 4.13: T-Flow **(a)**

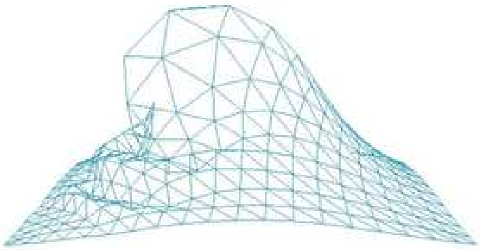


Figure 4.16: T-Flow **(a)**

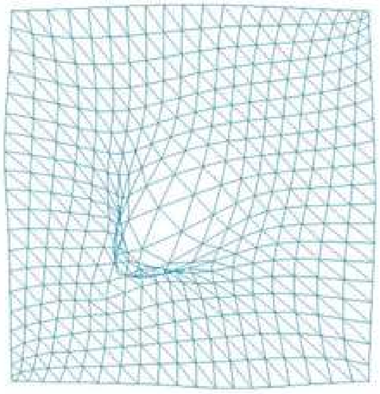


Figure 4.14: T-Flow **(b)**

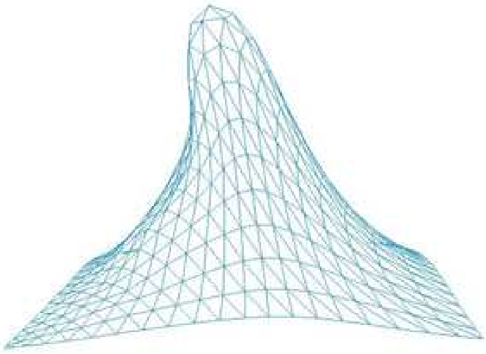


Figure 4.17: T-Flow **(b)**

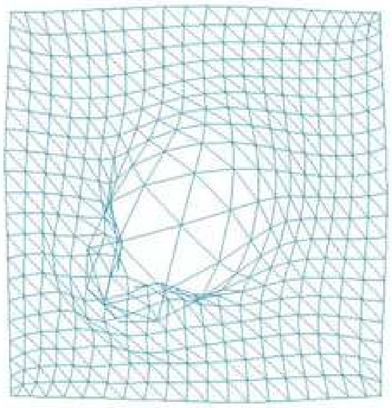


Figure 4.15: T-Flow **(c)**

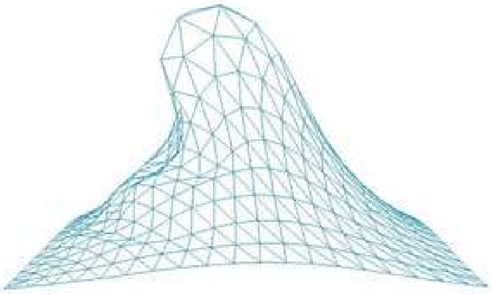


Figure 4.18: T-Flow **(c)**

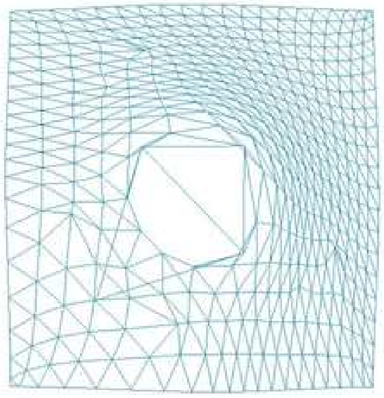


Figure 4.19: T-Flow **(d)**

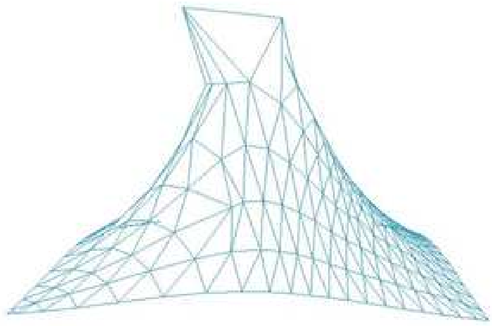


Figure 4.22: T-Flow **(d)**

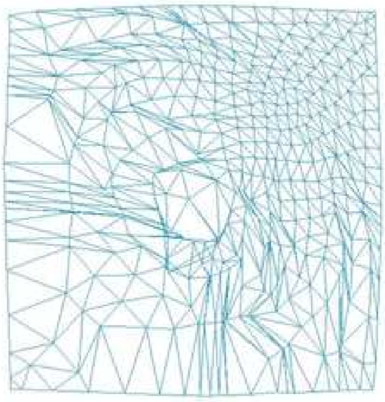


Figure 4.20: T-Flow **(e)**

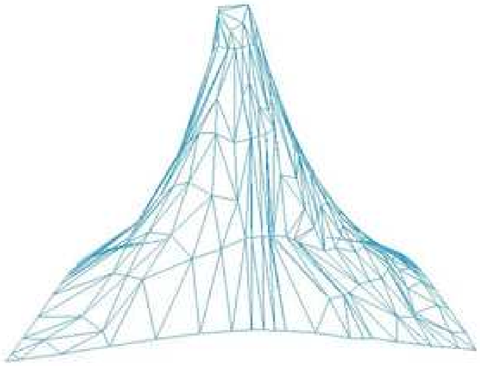


Figure 4.23: T-Flow **(e)**

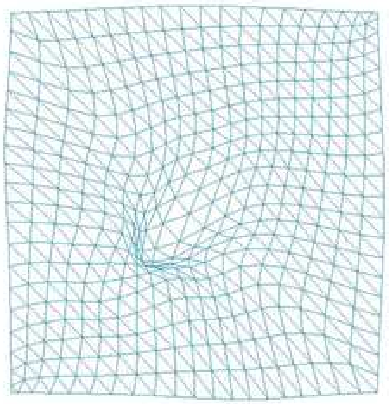


Figure 4.21: T-Flow **(f)**

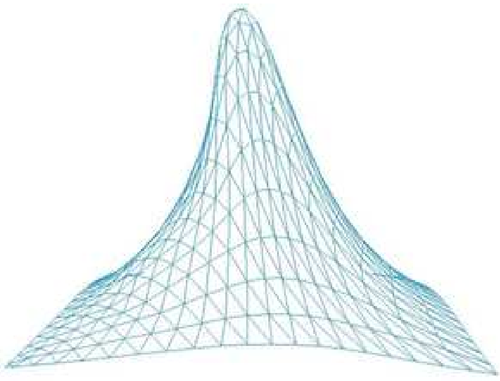


Figure 4.24: T-Flow **(f)**

Proposition:[Drift Phenomenon] *If $K < 0$ and the vertex distribution is sparse for t_{min} (dense for t_{max}) direction than t_{max} (t_{min}) direction then the vertices converge to the ridge which is corresponded k_{min} (k_{min}) by the tangent flow via the projection of the umbrella operator onto the tangent plane. The bi-umbrella operator is inherited this drift phenomenon from the umbrella operator. See Figures from 4.25 to 4.36, the thick dotted lines in right Figures represent bifurcations of the flow vectors.*

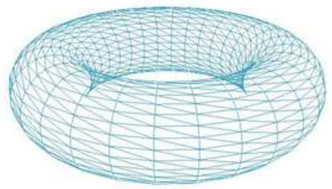


Figure 4.25: Original

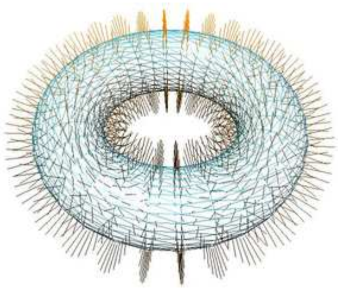


Figure 4.29: Normal

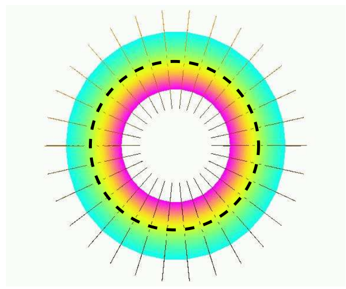


Figure 4.33: Normal

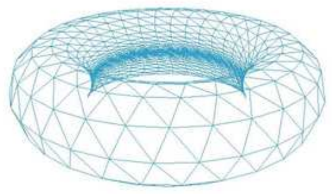


Figure 4.26: T-Flow **(a)**

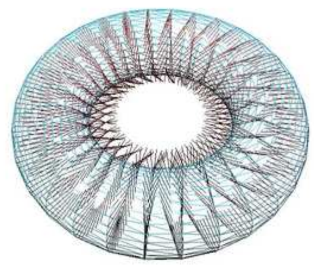


Figure 4.30: Umbrella

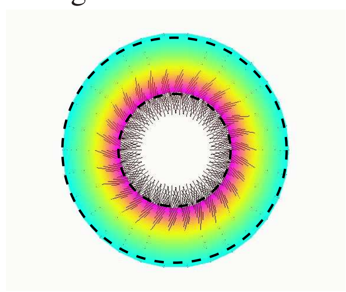


Figure 4.34: Umbrella

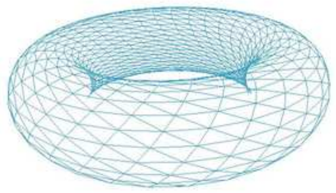


Figure 4.27: T-Flow **(b)**

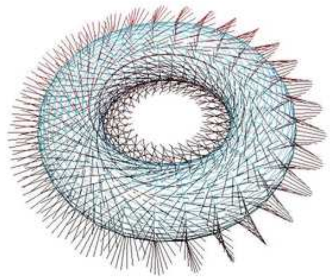


Figure 4.31: Bi-Umbrella

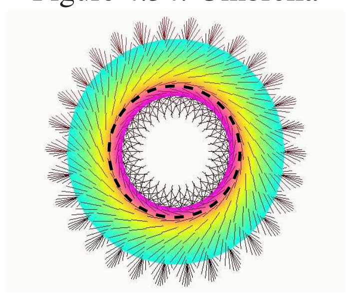


Figure 4.35: Bi-Umbrella

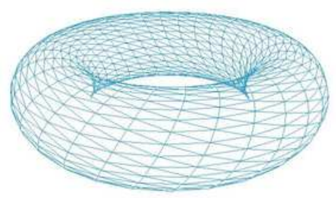


Figure 4.28: T-Flow **(f)**

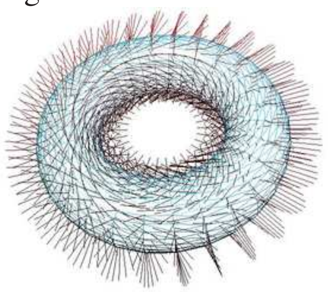


Figure 4.32: $\triangle^2_{B_{Area}}()$

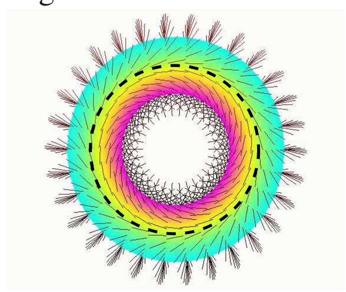


Figure 4.36: $\triangle^2_{B_{Area}}()$

Mesh Regularization Example: The following Filgures 4.40, 4.41 and 4.42 represent the reparametrization result of the tri-mountain via the proposed new tangent flow **(f)** 600000 iterations which show that the area weighted bi-Laplace-Beltrami tangent flow is useful for the vertex uniformization and mesh regularization of the triangulated polygonal surfaces with the non-uniform vertex distribution.

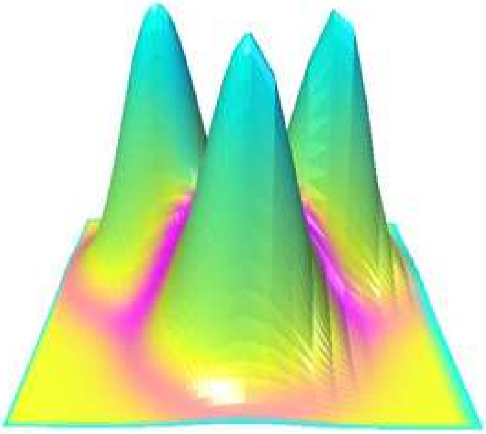


Figure 4.37: Original Meshes

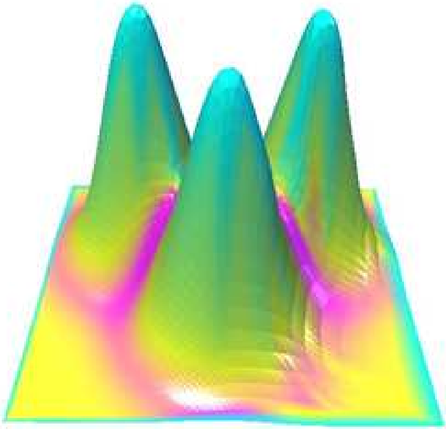


Figure 4.40: T-Flow **(f)**

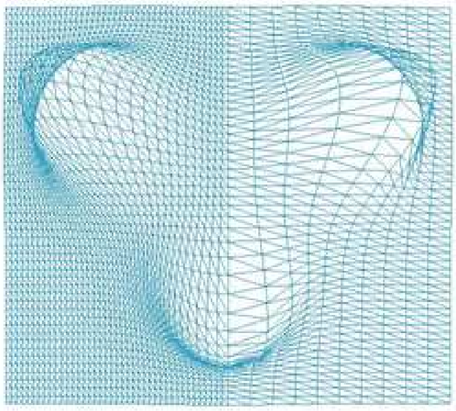


Figure 4.38: Original Meshes

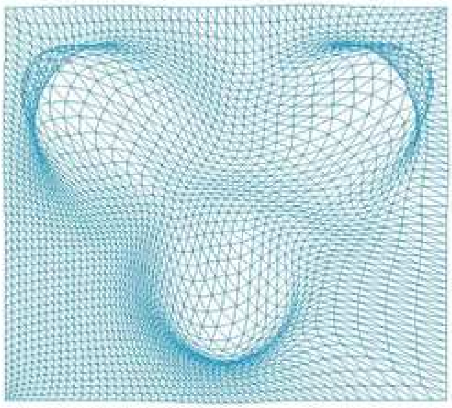


Figure 4.41: T-Flow **(f)**

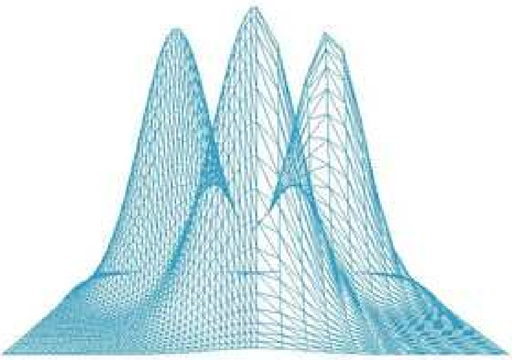


Figure 4.39: Original Meshes

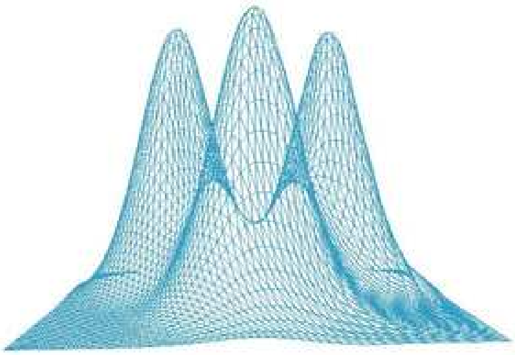


Figure 4.42: T-Flow **(f)**

4.3 Fair Surface Modeling via Elastic Flows

The elastic flows are implemented in this Section for the fair surface modeling [11, 31, 36, 37, 38]. The elastic flow is a surface diffusion similar to the smoothing with higher derivative propagation than the mean curvature flow, and the fair surface modeling is an interpolation of the prescribed boundaries method to satisfy the given boundary conditions via the represented elastic flows. The method is constructed by the alternate iterations of the elastic flow and regular subdivision. The elastica flow **(e)** is the best method in implemented five elastic flows from the numerical experiments.

Boundary Conditions: The boundary condition is given by the fixed boundary strips. It is equivalent to give the boundary normals, curvatures or tangents of boundaries. The limit surface interpolates these boundary conditions.

4.3.1 Boundary Strip Fairing

For the subdivision operations, the boundary strip is modified by the following new boundary strip fairing technique. First, the triangle is subdivided by the regular subdivision whose inserted new position of the vertices are located on the centers of the edges. Second, the following curve fairing is applied to the new boundary vertices (new inserted boundary vertex) and inner vertices (old inserted strip vertex). The old inserted strip vertex is the inserted into the old inner boundary strip (old strip vertex). Moreover, the original vertices (inner vertex, old strip vertex and old boundary vertex) are fixed. Finally, the middle vertices (new strip vertex) are moved to new positions which are determined by the arithmetic mean of the neighbor vertices without new strip vertices. The following Figure 4.43 denotes the classification of vertices. This is a new method to fair the boundaries.

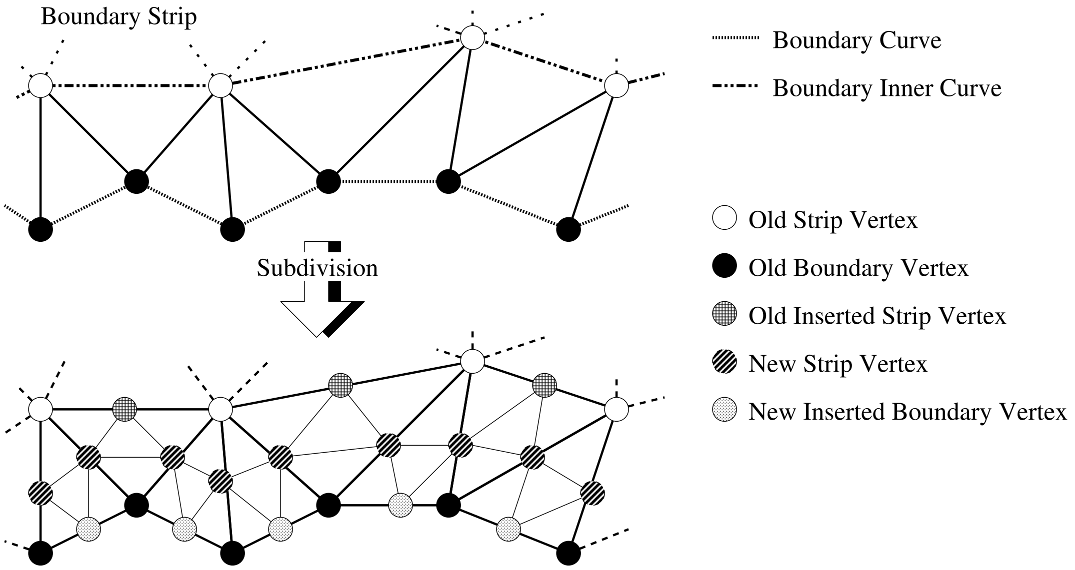


Figure 4.43: Boundary Strip Vertices Classification

Curve Fairing: The curve elastic flow is approximated by the following algebraic operations:

$$\mathbf{P}_{new} = \mathbf{P}_{old} + \frac{1}{2}L_{curve}(L(\mathbf{P}_{old})), \tag{4.18}$$

where $L_{curve}()$ operator represents the central difference of the successive vertices of the curve in equation (4.16). For one subdivision, this flow is applied by ten iterations. This operator propagates the umbrella operator of the surface with curve which is an approximation of 3D clothoid (liner curvature distribution curve [1, 40]) roughly.

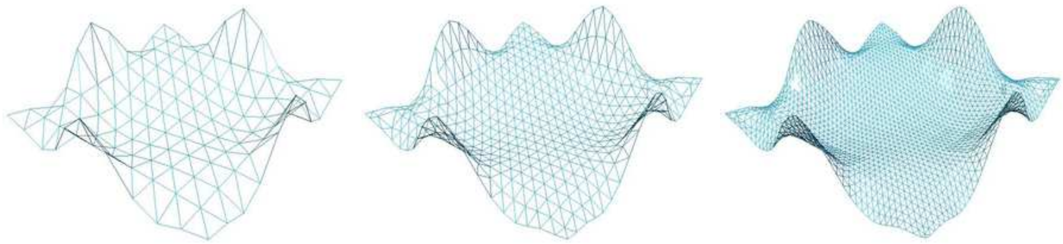


Figure 4.44: Example of Subdivision and Boundary Strip Fairing

The left figure model is an original polygonal surface. The center and right figures represent the boundary strip fairing models where the subdivision and the boundary strip fairing were applied once and twice, respectively.

For the numerical stability, the tangent flow is used via the area weighted bi-Laplace-Beltrami onto the tangent plane (see T-Flow **(f)** in Section 4.2) whose speed vector is denoted by $\epsilon_2 F\mathbf{T}$, where $\epsilon_2 = 2.4\frac{\mathbf{min}(A)}{2}$ (experimental value).

4.3.2 Bi-Laplacian Flow (a)

The bi-Laplacian flow is implemented by the algebraic operations:

$$\mathbf{P}_{new} = \mathbf{P}_{old} - \epsilon L^2(\mathbf{P}_{old}), \tag{4.19}$$

where ϵ must satisfy an inequality $0 < \epsilon \leq \frac{1}{2}$ because this is a bi-harmonic equation [33]. The bi-Laplacian flow propagates neighbor Laplacian values which are approximated by the umbrella operator.

4.3.3 Bi-Laplace-Beltrami Flow

The bi-Laplace-Beltrami flow is implemented by the following two types of the Laplace-Beltrami operators. These are new methods to fair the triangle meshes.

Bi-Laplace-Beltrami Flow (b):

$$\mathbf{P}_{new} = \mathbf{P}_{old} - \epsilon_1 \triangle^2_{B_{Des}}(\mathbf{P}_{old}) + \epsilon_2 F\mathbf{T}, \tag{4.20}$$

where $\epsilon_1 = \frac{\mathbf{min}(A)^2}{150}$.

Area Weighted Bi-Laplace-Beltrami Flow (c):

$$\mathbf{P}_{new} = \mathbf{P}_{old} - \epsilon\triangle^2_{B_{Area}}(\mathbf{P}_{old}), \tag{4.21}$$

where $\epsilon = 6\mathbf{min}(A)$ from the numerical experiments.

4.3.4 Laplace-Beltrami of Mean Curvature Flow (d)

The Laplace-Beltrami operator of the mean curvature flow **LBoMC** [36, 37] is implemented by the algebraic operations:

$$\mathbf{P}_{new} = \mathbf{P}_{old} - \epsilon_1\triangle_{B_{Des}}(\frac{1}{2}(\triangle_{B_{Des}}(\mathbf{P}_{old}) \cdot \mathbf{N})) + \epsilon_2 F\mathbf{T}, \tag{4.22}$$

where $\epsilon_1 = \frac{\mathbf{min}(A)^2}{24}$. This flow propagates neighbor mean curvature values, and achieves the linear distribution of the mean curvature for the limit surface.

4.3.5 Elastica Flow (e)

The curvature approximation method is a very important, when the numerical implementation is applied to the elastica flow which is defined in Section 2.2.3. Because the elastica flow has a nonlinear term $2H(H^2 - K)$ which is constructed by the mean and Gaussian curvatures, simultaneously. If the scale between the mean curvature and Gaussian curvature approximations is a very different scale, the nonlinear term $(H^2 - K)$ can take a minus sign: $(H^2 - K) < 0$. However, the smooth continuous surface must satisfy $(H^2 - K) \geq 0$, because the principal curvatures which have the term $\sqrt{H^2 - K}$ must be the real numbers. Therefore the curvature approximation methods which were represented in Section 3.5 except the modified Desbrun curvature approximation method can not be applied to the discrete elastica flow implementations.

The elastica flow is implemented by the algebraic operations:

$$\mathbf{P}_{new} = \mathbf{P}_{old} - \epsilon_1(\triangle_B(H) + 2H(H^2 - K))\mathbf{N} + \epsilon_2 F\mathbf{T}, \tag{4.23}$$

where $\epsilon_1 = \frac{\mathbf{min}(A)^2}{150}$, and the following curvature approximation method (see Section 3.5.1) is applied to this flow.

$$\triangle_B(H) \approx \triangle_{B_{Des}}(\frac{1}{2}\triangle_{B_{Des}}(\mathbf{P}_{old}) \cdot \mathbf{N}), \tag{4.24}$$

$$H \approx \frac{1}{2}\triangle_{B_{Des}}(\mathbf{P}_{old}) \cdot \mathbf{N}, \quad K \approx \lambda K_{Des}, \quad K_{Des} = \frac{3}{A}(2\pi - \sum_{i=1}^{M}\varphi_i), \tag{4.25}$$

$$\lambda = \min_{all\ points}\{\frac{H^2}{K_{Des}} : K_{Des} > H^2,\ H \neq 0\}. \tag{4.26}$$

This flow propagates neighbor squared integral of the normal curvature values, and achieves the minimum bending energy of the surface. This is a new method to fair the triangle meshes.

4.3.6 Comparison of Methods

The following Figures describe the elastic flows whose notations **(a)**, **(b)**, **(c)**, **(d)** and **(e)** correspond to the following Table 4.3. The boundary strips are fixed. The left, center and right figures represent meshes which are applied the elastic flows 3 times twice once, respectively.

Notation	Flow Name	Equation No.	Section
Flow **(a)**	Bi-Laplacian	3.11, 4.19	4.3.2
Flow **(b)**	Bi-Laplace-Beltrami	3.7, 4.20	4.3.3
Flow **(c)**	Area Weighted Bi-Laplace-Beltrami	3.9, 4.21	4.3.3
Flow **(d)**	**LBoMC**	3.7, 4.22	4.3.4
Flow **(e)**	Elastica [Best]	3.7, 4.23∼4.26	4.3.5

Table 4.3: Elastic Flow Methods

In the fairing cone model results, the bi-Laplacian **(a)** and area weighted bi-Laplace-Beltrami **(c)** flows develop the mean curvature discontinuity at top of the cone models (see Figures 4.50, 4.52, 4.55 and 4.57). The limit polygonal surface is almost hemisphere via the **LBoMC** flow **(d)** because of their linear mean curvature distribution (see Figure 4.58). The limit surfaces of the elastica **(e)** and bi-Laplace-Beltrami **(b)** flows are deformed by the swelling larger and smaller than the **LBoMC** flow **(d)**, respectively (see Figures 4.57, 4.58 and 4.59) because of their energies (see Chapter 2).

In the virtual water pipe model fairing results, the bi-Laplace-Beltrami **(b)** and **LBoMC (d)** flows develop singularities which are the fatal error for the fairing surface modeling (see Figures 4.61, 4.63, 4.66, 4.68, 4.71 and 4.73). Again, the bi-Laplacian **(a)** and area weighted bi-Laplace-Beltrami **(c)** flows develop the mean curvature discontinuity at the neighborhood of boundaries (see Figures 4.65, 4.67, 4.70 and 4.72). Proposed new discrete elastica flow **(e)** provides us good results about this model (see Figures 4.76, 4.64, 4.69 and 4.74).

All methods achieve fairing for the triangulated polygonal surface. The bi-Laplacian flow **(a)** has the highest flow speed and highest stability of numerical scheme in the considered elastic flows, but the vertex distribution dependency is a very high. The bi-Laplace-Beltrami flow **(b)** has the high flow speed and high stability of numerical scheme, but the limit surface has the high bending energy because of its energy functional and Euler-Lagrange equation. The area weighted bi-Laplace-Beltrami flow **(c)** has the low flow speed and low stability of numerical scheme in the considered elastic flows. The **LBoMC** flow **(d)** has the low flow speed and low stability of numerical scheme, but the limit surface has the low bending energy and linear mean curvature distribution. The elastica flow **(e)** has the low flow speed and low stability of numerical scheme, but the the limit surface has the minimum bending energy in the considered elastic flows. The elastica flow may not develop the surface singularities and polygonal singularities (using tangent flow).

The numerical experiments show that the elastica flow **(e)** is the best method for the fair surface modeling via the surface flow in considered flows.

Cone Fairing via Bipolar Color Mapping

Figure 4.45: Flow **(a)**

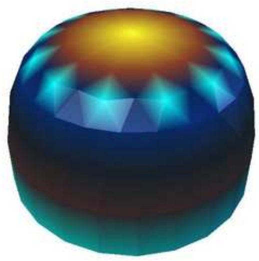


Figure 4.50: Flow **(a)**

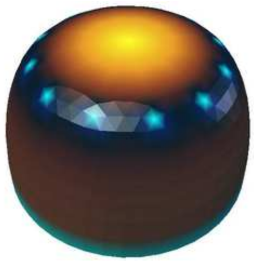


Figure 4.55: Flow **(a)**

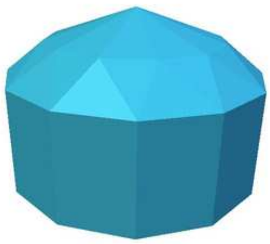


Figure 4.46: Flow **(b)**

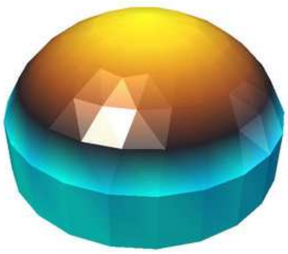


Figure 4.51: Flow **(b)**

Figure 4.56: Flow **(b)**

Figure 4.47: Flow **(c)**

Figure 4.52: Flow **(c)**

Figure 4.57: Flow **(c)**

Figure 4.48: Flow **(d)**

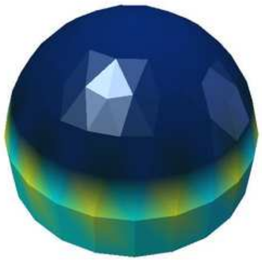


Figure 4.53: Flow **(d)**

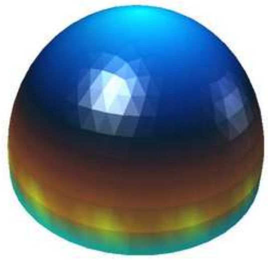


Figure 4.58: Flow **(d)**

Figure 4.49: Flow **(e)**

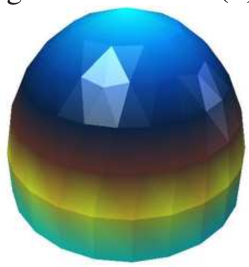


Figure 4.54: Flow **(e)**

Figure 4.59: Flow **(e)**

Water Pipe Model Fairing via Pastel Color Mapping

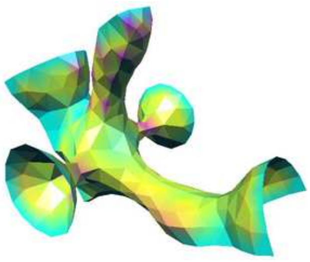


Figure 4.60: Flow **(a)**

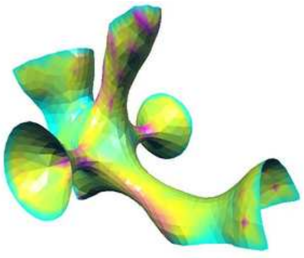


Figure 4.65: Flow **(a)**

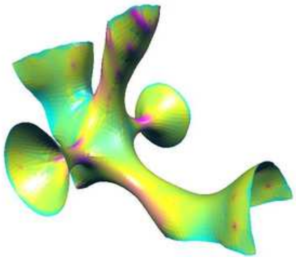


Figure 4.70: Flow **(a)**

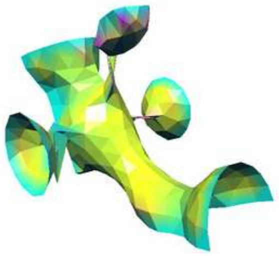


Figure 4.61: Flow **(b)**

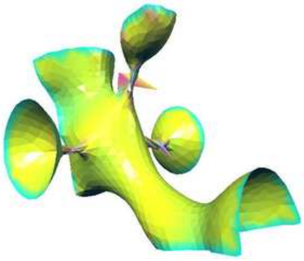


Figure 4.66: Flow **(b)**

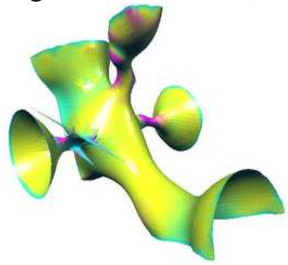


Figure 4.71: Flow **(b)**

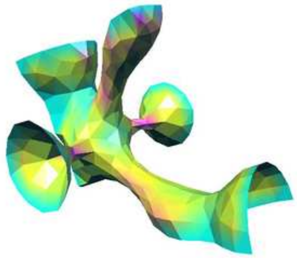


Figure 4.62: Flow **(c)**

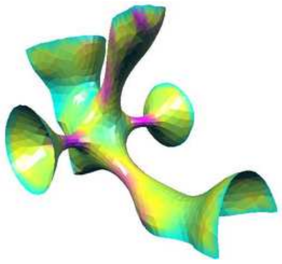


Figure 4.67: Flow **(c)**

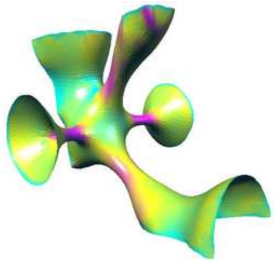


Figure 4.72: Flow **(c)**

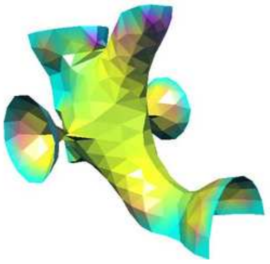


Figure 4.63: Flow **(d)**

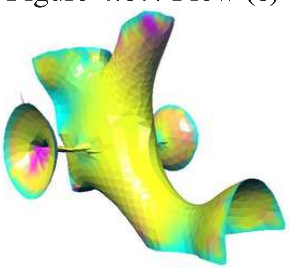


Figure 4.68: Flow **(d)**

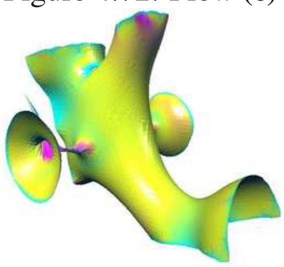


Figure 4.73: Flow **(d)**

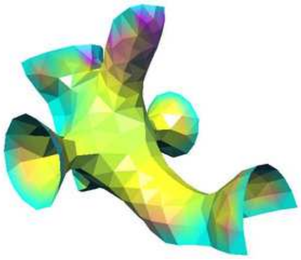


Figure 4.64: Flow **(e)**

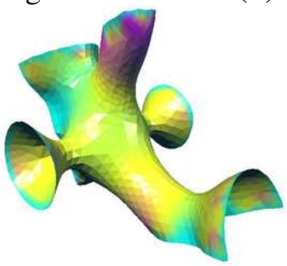


Figure 4.69: Flow **(e)**

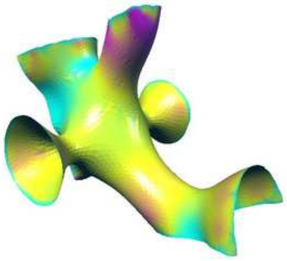


Figure 4.74: Flow **(e)**

Water Pipe Model Fairing via Pastel Color Mapping

Figure 4.75: Original Water Pipe Model

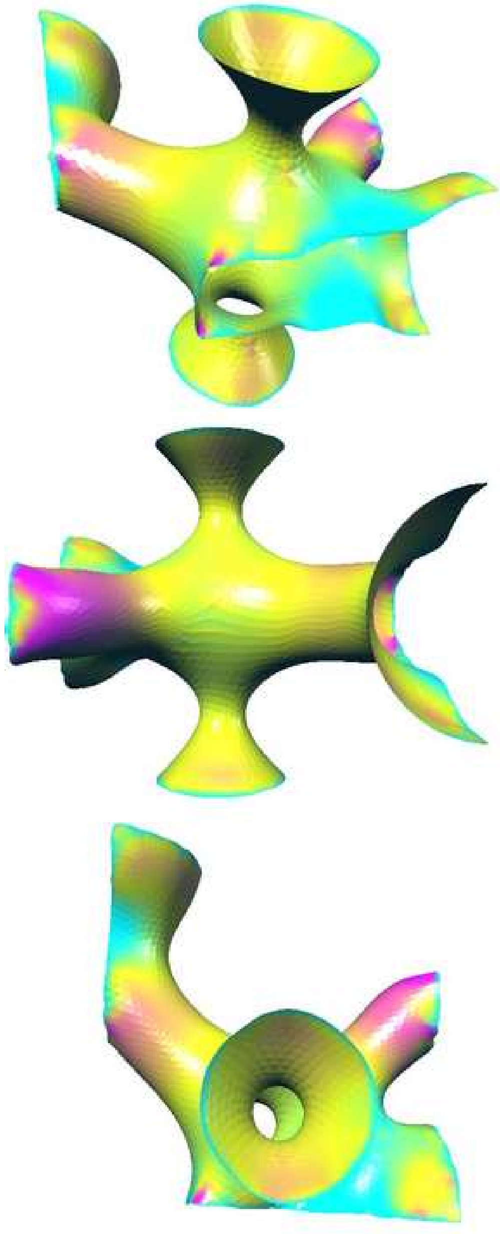


Figure 4.76: Fairing Water Pipe Model via Elastica Flow **(e)**

The left figures show the original water pipe mesh. The right figures represent the fairing water pipe mesh via the elastica flow **(e)**. *The original water pipe mesh is modeled by the set-theoretic operations of the implicit functions. The fairing water pipe mesh can be evaluated as a good mesh by the color distribution and their shape. Because the color of figures describe the mean curvature distribution which inspires the discrete smoothness and fairness with a human perception.*

4.3.7 Fair Surface Modeling Example via Elastica Flow

The possible application of fair surface modeling is the following fill hole problem. When the object is scanned by a laser scanner, a part of the surface information which has high specular parts can be lost. Because the laser is scattered (diffused) by these high specular parts. Also the accurate reflection of the laser is not performed. Generally, the scanner software misrecognizes such kind of a part to be as a hole. The following numerical experiments show that the elastica flow is useful for the shape restoration after filling a hole by the fair surface for the example of the fair surface modeling application.

Original Holed Bunny: First, the following Figures represent original bunny surface with a hole which can be confirmed by the black color of the bunny surface.

Initial Filled Plate: Next, the hole is filled up by a flatness surface which is created for example the Delaunay triangulation [14, 28].

Interpolated Bunny: Finally, the elastica flow is applied to the initial filled model.

Chapter 5

Conclusion

The thesis devoted the surface modeling approaches based on the geometric variational concepts with curvature-driven shape deformations. Instead of considering smooth continuous surfaces, the triangulated polygonal surfaces are used in the implementations. In Chapter 2, the membrane and thin plate energies were considered on the surface metric, and corresponding surface flows and its approximation flows were introduced. The singularity analysis of the analytical surface flows were applied to three simple shapes. The elastica flow is more stable than the mean curvature, bi-Laplace-Beltrami, Laplace-Beltrami of mean curvature flows from the results of flow direction considerations with local shape analysis. The discrete curvature approximation was developed via the Laplace-Beltrami operator with modified Desbrun method in 3. Three important surface modeling applications which are the discrete stable surface smoothing via the area weighted Laplace-Beltrami flow, the mesh regularization via the area weighted bi-Laplace-Beltrami tangent flow, and the fair surface modeling via the elastica flow were considered and implemented in 4.

5.1 Main New Results

The thesis proposed the new methods accompanied by the following main results.

Curvature Approximation: The Laplace-Beltrami operator with modified Desbrun method is the best curvature estimation method in implemented six methods about the irregular triangle meshes. The modified Desbrun method was constructed by scaling of the area weighted angle deficit method to satisfy the inequality $H^2 - K \geq 0$ where H and K are the surface mean and Gaussian curvatures, respectively.

Surface Smoothing: The discrete stable triangulated polygonal surface denoising was described by the area weighted Laplace-Beltrami flow whose operator is the area weighted normalized umbrella operator. The numerical experiments indicated that the proposed new method combine the excellent stability properties of the umbrella operator with remarkable geometric properties of the mean curvature normal vector.

Mesh Regularization: The mesh regularization was designed by the area weighted bi-Laplace-Beltrami tangent flow for the irregular vertex distribution triangle meshes whose speed vector is constructed by the projection of squared area weighted normalized umbrella operator onto the tangent plane. The numerical experiments show that the proposed new method well diffuse uniformity vertices to preserve geometry for the same geometry parts.

Fair Surface Modeling: The fair surface was modeled by the regular subdivision with the elastica flow whose speed function minimizes a squared normal curvature energy. The method was implemented by the proposed new modified Desbrun curvature approximation method with the tangent flow. The tangent flow avoided polygonal singularities. The elastica flow did not develop singularities which were discussed in Chapter 2. The numerical experiments represented that the proposed new discrete elastica flow is very useful for the fair surface modeling of the triangle meshes with interpolating boundary conditions.

5.2 Program Implementation

All methods were implemented by Java Development TooKit 1.3 and Java3D application programming interface developed by Sun Microsystems. A few mathematical analysis was calculated by Mathematica 4.0 developed by Wolfram Research. These programs were executed on IRIX 6.5 O2 and Solaris 2.6 Ultra workstations developed by Silicon Graphics and Sun Microsystems, respectively.

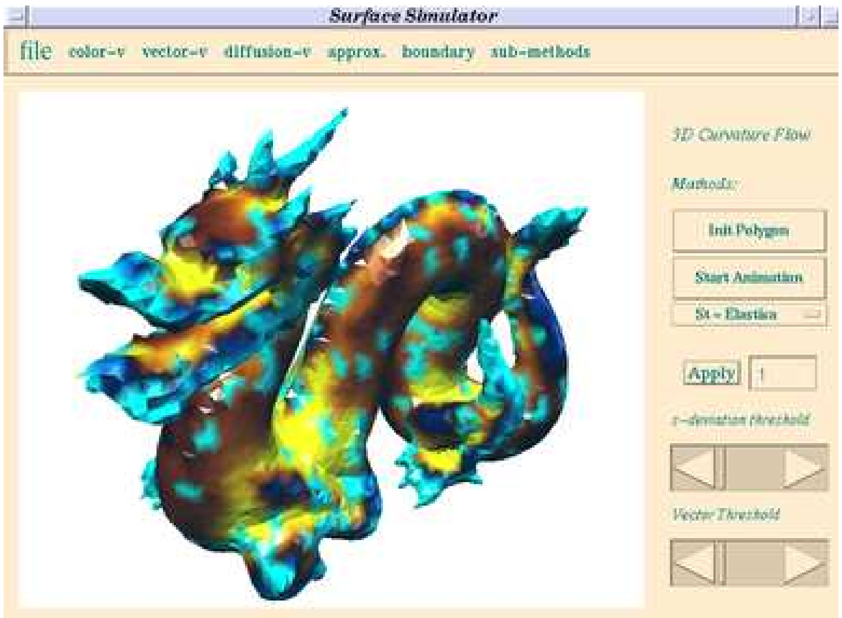


Figure 5.1: Surface Flow Simulator

5.3 Future Works

The following future works can be considered.

- The current surface flow speeds are slow speeds, and multi-grid approach is implemented manually then the automatic adaptive mesh implementation will be applied to the program. The other numerical methods will be considered to solve the Euler-Lagrange equations for example signal processing approaches, direct orthonormal basis methods and gradient decent methods for the calculation speed up.

- Almost time step-size of nonlinear flows were experimental values (consideration from the heat equation) than the strict mathematical analysis will be applied to the time step-size and tangent flows.

- The Euler-Lagrange equation of more higher derivative energies will be considered for example the Euler-Lagrange equation of the following popular energy functional [16]:

$$\iint_{\Omega} (\frac{\partial k_{max}}{\partial t_{max}})^2 + (\frac{\partial k_{min}}{\partial t_{min}})^2 \, dA,$$

  where k_{max} and k_{min} are the principal curvatures, and t_{max} and t_{min} are the principal directions.

- Other surface modeling approximations will be implemented. The feature preserving or enhancement smoothing will be considered by the nonlinear diffusion of the normal vector with ridge and ravine weights. The reparameterization of the polygonal surfaces which converges vertices to surface feature curves with preserving geometry will be implemented via the discrete gradient tangent flow. The modification methods of the polygonal surface will be applied via the elastica flow with elastic net algorithms.

References

[1] A. G. Belyaev, E. V. Anoshkina, S. Yoshizawa, and M. Yano, "Polygonal curve evolutions for planar shape modeling and analysis", *International Journal of Shape Modeling*, Vol. 5, No. 2, 1999, pp. 195-217.

[2] A. G. Belyaev, A. A. Pasko, and T. L. Kunii, "Ridges and Ravines on Implicit Surfaces", *Computer Graphics International '98 Proceedings*, 1998, pp. 530-535.

[3] A. G. Belyaev, E. V. Anoshkina, and T. L. Kunii, "Ridges, Ravines, and Singularities", *Topological Modeling for Visualization*, Chapter 18 in A. T. Fomenko and T. L. Kunii, Springer-Verlag, 1997.

[4] A. G. Belyaev and S. Yoshizawa, "On Evolute Cusps and Skeleton Bifurcations", accepted to SMI'01.

[5] A. Gray, *Modern Differential Geometry of Curves and Surfaces With Mathematica*, 2nd edition, CRC Press, 1998.

[6] A. Vakhutinsky and B. L. Golden, "A Hierarchical Strategy for Solving Traveling Salesman Problems", *Journal of Heuristics*, Vol. 1, 1995, pp. 67-76.

[7] C. Lu, Y. Cao and D. Mumford, "Surface Evolution under Curvature Flows", Submitted for publication, Available from "http://www.dam.brown.edu/people/mumford/research_new.html".

[8] D. J. Struik, *Lectures on Classical Differential Geometry*, 2nd edition, Dover Publications, 1988.

[9] D. L. Chopp and J. A. Sethian, "Motion by Intrinsic Laplacian of Curvature", *Interfaces and Free Boundaries*, 1999, pp. 1-18.

[10] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, 4nd edition, Academic Press Ltd, 1999.

[11] G. Taubin, "A Signal Processing Approach to Fair Surface Design", *SIGGRAPH'95 Proceedings*, 1995, pp. 315-358.

[12] G. Taubin, "Estimating the Tensor of Curvature of a Surface from a Polyhedral Approximation", *Fifth International Conference on Computer Vision*, 1995, pp. 902-907.

[13] G. Taubin, “Optimal Surface Smoothing as Filter Design”, *Fourth European Conference on Computer Vision*, 1996.

[14] H. Edelsbrunner, *Algorithms in Combinatorial Geometry*, Japanese Edition, Kyoritsu Shuppan, 1995.

[15] H. Hoppe, “Progressive Meshes”, *SIGGRAPH’97 Proceedings*, 1997, pp. 99-108.

[16] H. P. Moreton and C. H. Sequin, “Functional Optimization for Fair Surface Design”, *SIGGRAPH’92 Proceedings*, 1992, pp. 167-176.

[17] J. A. Sethian, *Level Set Methods*, Cambridge University Press, 1996.

[18] J. Hoschek, U. Dietz, and W. Wilke “A Geometric Concept of Reverse Engineering of Shapes: Approximation and Feature Lines”, *Mathematical Methods for Curves and Surfaces II*, Vanderbilt University Press, 1998, pp. 253-262.

[19] J. Kaasa and G. Westgaard, “Analysis of Curvature-Related Surface Shape Properties”, *Curves and Surfaces with Applications in CAGD*, pp. 211-215 in A. L. Mehaute, C. Rabut, and L. L. Schumaker, Vanderbit University Press, 1997.

[20] J. Vollmer, R. Mencl, and H. Muller “Improved Laplacian Smoothing of Noisy Surface Meshes”, *Proceedings of EUROGRAPHICS ’99: Computer Graphics Forum*, Vo. 18, No. 3, 1999, pp. 131-138.

[21] J. W. Wesselink, “Variational Modeling of Curves and Surfaces”, *PhD thesis, Eindhoven University of Technology*, 1996.

[22] K. Brakke, “The Surface Evolver”, *Experimental Mathematics*, Vol. 1, No. 2, 1992, pp. 141-165.

[23] K. Watanabe and A. G. Belyaev, “Detection of Curvature Extrema on Polygonal Surfaces”, *Human and Computer 2000 Proceedings*, 2000, pp. 11-14.

[24] L. Euler, Additamentum “De Curvis Elasticis”, *Methodus Inverniendi Lineas Curvas Maximi Minive Proprietate Gaudentes*, Lausanne, 1744.

[25] L. Hsu, R. Kusner, and J. Sullivan, “Minimizing the Squared Mean Curvature Integral for Surfaces in Space Forms”, *Experimental Mathematics*, Vol. 1, No. 3, 1992, pp. 191-207.

[26] L. Kobbelt, S. Campagna, J. Vorsatz, and H. P. Seidel, “Interactive Multi-Resolution Modeling on Arbitrary Meshes”, *SIGGRAPH’98 Proceedings*, 1998, pp. 105-114.

[27] L. Kobbelt, M. Botsch, K. Kahler, C. Rossl, R. Schneider, and J. Vorsatz, “Geometric Modelling Based on Polygonal Meshes”, *EUROGRAPHICS 2000 Tutorial*, T4, 1998.

[28] M. D. Berg, M. V. Kerveld, M. Overmars and O. Schwarzkopf, *Computational Geomerty: Algorithms and Applications*, Springer-Verlag Berlin Heidelberg, 1997.

[29] M. E. Taylor, *Partial Differential Equations II: Qualitative Studies of Linear Equations*, Applied Mathematical Sciences 116, Springer-Verlag New York, 1996.

[30] M. E. Taylor, *Partial Differential Equations III: Nonlinear Equations*, Applied Mathematical Sciences 117, Springer-Verlag New York, 1996.

[31] M. Desbrun, M. Meyer, P. Scrhoder, and A. H. Barr, "Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow", *SIGGRAPH'99 Proceedings*, 1999, pp. 317-324.

[32] M. Desbrun, M. Meyer, P. Schroder, and A. H. Barr, "Discrete Differential-Geometry Operators in nD", Submitted for publication, Available from "http://www.multires.caltech.edu/pubs".

[33] M. Giaquinta and S. Hildebrandt, *Calculus of Variations I*, Springer-Verlag Berlin Heidelberg, 1996.

[34] M. Hosaka, *Modeling of Curves and Surfaces in CAD/CAM*, Springer-Verlag, 1996.

[35] R. Kimmel, R. Malladi, and N. Sochen, "Image as Embedding Maps and Minimal Surfaces: Movies, Color, and Volumetric Medical Images", *International Journal of Computer Vision*, 2000, Vol. 32, No. 2, pp. 111-129.

[36] R. Schneider and L. Kobbelt, "Generating Fair Meshes with G^1 Boundary Conditions", *Geometric Modeling and Processing 2000 Proceedings*, 2000, pp. 251-260.

[37] R. Schneider and L. Kobbelt, "Geometric Fairing of Irregular Meshes for Free-Form Surface Design", Computer Aided Geometric Design, to appear.

[38] R. Schneider and L. Kobbelt, "Discrete Fairing of Curves and Surfaces based on Linear Curvature Distribution", *Curve and Surface Design: Saint-Malo*, 1999, pp. 371-380.

[39] R. Weinstock, *Calculus of Variations: With Applications to Physics and Engineering*, Dover Publications, 1974.

[40] S. Yoshizawa, "Nonlinear Spline Modeling with Curvature-Driven Deformations", *The Journal of Three Dimensional Images*, Vol. 12, No. 3 1998, pp. 53-58.

[41] S. Yoshizawa, "Stable Discrete 2D and 3D Curvature Flows", *Human and Computer 2000 Proceedings*, 2000, pp. 87-91.

[42] S. Yoshizawa and A. G. Belyaev, "Visualization and Study of Dynamic 2D Shapes via Curvature", *International Conference on Multimedia Modeling (MMM 2000) Proceedings*, 2000, pp. 469-489.

[43] U. Clarenz, U. Diewald, and M. Rumpf, "Anisotropic Geometric Diffusion in Surface Processing", *IEEE Visualization 2000*, 2000, pp. 397-405.

[44] V. I. Smirnov, *Integral Equations and Partial Differential Equations*, A Course of Higher Mathematics Volume IV, Addison-Wesley Publishing Company, 1964.

[45] W. Welch and A. Witkin, "Free-Form Shape Design Using Triangulated Surfaces", *SIGGRAPH'94 Proceedings*, 1994, pp. 247-256.

[46] W. Welch and A. Witkin, "Variational Surface Modeling", *SIGGRAPH'92 Proceedings*, 1992, pp. 157-166.

[47] W. Welch, "Topological Design for Variational Curves and Surfaces", *Ph.D. dissertation, School of Computer Science, Carnegie Mellon University*, 1995.

[48] Y. Ohtake, A. G. Belyaev, and I. Bogaevski, "Polyhedral Surface Smoothing with Simultaneous Mesh Regularization", *Geometric Modeling and Processing 2000 Proceedings*, 2000, pp. 229-237.

[49] Y. Ohtake, A. G. Belyaev, and A. Pasko, "Dynamic Meshes for Accurate Polygonization of Isosurfaces with Sharp Features", Accepted to SMI'01.

[50] Y. Giga and Y.-G. Chen, *Analysis of Evolving Surfaces*, Nihon-hyouron, 1996 (In Japanese).

[51] Z. J. Wood, M. Desbrun, P. Schneider, and D. Green, "Semi-Regular Mesh Extraction from Volumes", *IEEE Visualization 2000*, 2000, pp. 275-282.

Appendix A

Minimizing Function of Mean and Gaussian Curvatures with Euler-Lagrange Equation

The useful formula was described in [33] about the Euler-Lagrange equation which corresponds to minimize the mean and Gaussian curvatures function $f(H,K)$:

$$\int\int_{\Omega} f(H,K)\,dA \Longrightarrow min.$$

The Euler-Lagrange equation of the above variational problem is given by

$$\Diamond f_K + \frac{1}{2}\triangle_B(f_H) + 2HKf_K + (2H^2 - K)f_H - 2fH = 0,$$

where f_H and f_K are the derivatives of $f(H,K)$ with respect to the mean and Gaussian curvatures H and K, respectively, and the metric tensor

$$g_{\alpha\beta} = \begin{pmatrix} E & F \\ F & G \end{pmatrix},$$

$$g = det(g_{\alpha\beta}) = EG - F^2,$$

$$g^{\alpha\beta} = \begin{pmatrix} E & F \\ F & G \end{pmatrix}^{-1} = \frac{1}{EG - F^2}\begin{pmatrix} G & -F \\ -F & E \end{pmatrix},$$

$$h^{\alpha\beta} = \begin{pmatrix} L & M \\ M & N \end{pmatrix}^{-1} = \frac{1}{LN - M^2}\begin{pmatrix} N & -M \\ -M & L \end{pmatrix}^{-1}.$$

The differential operators $\Diamond$ and $\triangle_B$ are defined by

$$\Diamond\phi = \frac{1}{\sqrt{g}}D_\alpha[\sqrt{g}Kh^{\alpha\beta}D_\beta\phi],$$

$$\triangle_B\phi = \frac{1}{\sqrt{g}}D_\alpha[\sqrt{g}g^{\alpha\beta}D_\beta\phi].$$

Appendix B

Taubin's Tensor of Curvature Estimation Method

The tensor of curvature estimation method was described in [12] which is based on the normal curvature $k_n(\theta)$ with the directional curvatures and principal directions:

$$k_n(\theta) = \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}^t \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}.$$

Let us show the numerical implementation of this method. The discrete tangent vectors are given by

$$\mathbf{T}_i = \frac{(\mathbf{P} - \mathbf{Q}_i)_{proj\mathbf{N}^\perp}}{|(\mathbf{P} - \mathbf{Q}_i)_{proj\mathbf{N}^\perp}|}.$$

The discrete curvature tensor matrix $\mathbf{R}$ is given by

$$\mathbf{R} = \frac{1}{A}\sum_{i=1}^{M} a_i k_i \mathbf{T}_i \mathbf{T}_i^t.$$

The Householder matrix $\mathbf{U}$ is given by

$$\mathbf{U} = \mathbf{I} - 2\mathbf{W}\mathbf{W}^t, \tag{B.1}$$

where

$$\mathbf{W} = \frac{\mathbf{e}_1 \pm \mathbf{N}}{|\mathbf{e}_1 \pm \mathbf{N}|}, \quad \mathbf{e}_1 = (1, 0, 0)^t,$$

with a minus sign if $\mathbf{e}_1 - \mathbf{N} > \mathbf{e}_1 + \mathbf{N}$, and plus sign otherwise. The discrete directional curvature tensor is given by

$$\mathbf{U}^t\mathbf{R}\mathbf{U} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & r_{11} & r_{12} \\ 0 & r_{21} & r_{22} \end{pmatrix}.$$

Now the rotation parameter

$$t_\theta = \frac{-r_{11} + r_{22}}{2r_{12}}, \quad t = \pm \frac{1}{|t_\theta| + \sqrt{t_\theta^2 + 1}}, \tag{B.2}$$

where $t_\theta < 0 \Longrightarrow t_\theta \leftarrow -t_\theta$. The eigenvalues of the curvature tensor matrix are given by the Givens rotation:

$$k_{t_1} = r_{11} - tr_{12}, \quad k_{t_2} = r_{22} - tr_{12},$$

$$k_1 = 3k_{t_1} - k_{t_2}, \quad k_2 = 3k_{t_2} - k_{t_1}.$$

The mean and Gaussian curvature H and K are approximated by

$$H \approx \frac{k_1 + k_2}{2}, \quad K \approx k_1 k_2.$$

Appendix C

Fundamental Forms and Curvatures

The appendix gives us notions and definitions of the fundamental forms and curvatures with basic knowledge of surface geometry. Let us consider a parametric surface $\mathbf{S}(u, v)$ and its unit normal vector $\mathbf{n}$. The first fundamental form of the surface $\mathbf{S}(u, v)$ is given by

$$\mathbf{I}(u, v) = Edu^2 + 2Fdudv + Gdv^2,$$

where $E = \mathbf{S}_u \cdot \mathbf{S}_u$, $F = \mathbf{S}_u \cdot \mathbf{S}_v$, and $G = \mathbf{S}_v \cdot \mathbf{S}_v$, and E, F and G are called the coefficients of the first fundamental form. The second fundamental form of the surface $\mathbf{S}(u, v)$ is given by

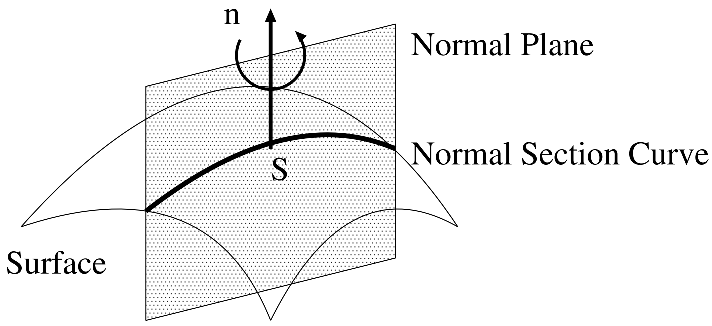
$$\mathbf{II}(u, v) = Ldu^2 + 2Mdudv + Ndv^2,$$

where $L = \mathbf{S}_{uu} \cdot \mathbf{n}$, $M = \mathbf{S}_{uv} \cdot \mathbf{n}$, and $N = \mathbf{S}_{vv} \cdot \mathbf{n}$, and L, M and N are called the coefficients of the second fundamental form.

The surface normal curvature $k_n(\mathbf{t}(u, v))$ is given by

$$k_n(\mathbf{t}) = \frac{\mathbf{II}(\mathbf{t}(u, v))}{\mathbf{I}(\mathbf{t}(u, v))}$$

for the normal plane which is spanned by the normal $\mathbf{n}$ and tangent $\mathbf{t}$. The surface normal curvature represents the curve curvature whose curve (normal section curve) is defined by the intersection of the surface and normal section. The maximum and minimum of the normal curvature are called the surface principal curvatures k_{max} and k_{min} in a revolution of $\mathbf{t}$ around $\mathbf{n}$. The corresponding principal curvature directions are called the principal directions.



The mean curvature H and Gaussian curvature K are given by

$$H = \frac{1}{2}(k_{max} + k_{min}), \quad K = k_{max}k_{min}$$

which can be defined by the coefficients of the first and second fundamental forms:

$$H = \frac{EN - 2FM + GL}{2(EG - F^2)}, \quad K = \frac{LN - M^2}{EG - F^2}.$$

The principal curvatures can be represented by the mean and Gaussian curvatures:

$$k_{max} = H + \sqrt{H^2 - K}, \quad k_{min} = H - \sqrt{H^2 - K},$$

where $H^2 - K > 0$ for an analytical surface in $\Re^3$.