Computer-Aided Design and Applications

http://www.cadanda.com

Visualizing High-Order Surface Geometry

Pushkar P. Joshi[1,2] and Carlo H. Séquin[2]

[1]Adobe Systems Inc., pushkarj@adobe.com
[2]University of California, Berkeley, sequin@cs.berkeley.edu

ABSTRACT

We have derived parameters that describe the higher-order geometric behavior of smooth surfaces. Our parameters are similar in spirit to the principal directions and principal curvatures that succinctly capture second-order shape behavior. We derive our parameters from a cylindrical Fourier decomposition around the surface normal. We present a visualization program for studying the influence of the various terms of different degrees on the shape of the local neighborhood of a surface point. We display a small surface patch that is controlled by two sets of parameters: One set is a simple polynomial description of the surface geometry in Cartesian coordinates. The other one is a set of Fourier components grouped by angular frequency and by their phase shifts. Manipulating the values in one parameter set changes the geometry of the patch and also updates the parameter values of the other set.

Keywords: principal curvature, higher-order shape parameters, Fourier analysis of geometry, interactive visualization.
DOI: 10.3722/cadaps.2009.263-268

1. INTRODUCTION

The geometric neighborhood of a point on a smooth surface is commonly studied and readily understood up to second order. Analysis of the zeroth-order (position) and first-order (tangent) shape needs only an elementary knowledge of geometry. Most courses and books on differential geometry give a detailed, visual exposition of second-order geometric concepts like mean curvature, Gaussian curvature, principal curvatures and their directions. As a result, up to second-order, the shape of a surface point can be intuitively expressed using simple geometric primitives. For example, we can characterize the second-order shape of a surface point as elliptic, parabolic, or hyperbolic simply by querying the sign of Gaussian curvature at that point. This characterization of geometric analysis has resulted in a widespread intuitive understanding of second-order shape behavior.

Beyond second order, an equivalent understanding of the shape of a surface point is rare. Researchers solve geometric problems involving higher-order terms by using a parameterization-dependent tensor analysis of the surface behavior. For example, to study surface behavior up to third order, Mehlum and Tarrou [5] build a function that yields the derivative of normal curvature in any specified direction. Similarly, Maekawa et al. [4] solve a cubic equation to produce the directions that have local maxima or minima in curvature variation. While such third-order tensor analysis yields accurate information for *specific* directions, it is difficult to obtain a quick, intuitive understanding of the *entire* third-order shape at a surface point in question.

The recent Ph.D. thesis by P. P. Joshi [2] shows that all the third-order shape behavior at a surface point can be succinctly expressed using only four geometrically intuitive, parameterization-independent parameters. In order to study the various parameters and to demonstrate their respective influence on surface geometry, we developed an interactive visualization program.

Our program displays a small surface patch centered at the point of analysis, and shows the different shapes achieved under the influence of different combinations of shape parameters. In particular we provide two different ways to specify the geometry of that surface patch. On the one hand, we describe it as a complete polynomial expression that includes all terms up to third order (an extension to higher-order terms is currently under way). Alternatively, the same geometry is described as a set of angular Fourier components around the surface normal at the given point. Either one of these two parameter sets can be adjusted interactively through a bank of sliders. As one set of parameter values is modified, the other set of parameter values is updated appropriately. The few users who have used this visualization program so far all have felt that it greatly enhanced their understanding of higher-order surface geometry.

This paper describes the visualization tool that we built to study high-order surface behavior and to observe the high-order basis shapes. The mathematical derivations that yield these basis shapes are presented in Chapter 3 ("Intuitive Exposition of Third-Order Surface Behavior") of the thesis [2].

2. FOURIER ANALYSIS

Our program currently displays shape behavior up to third order. We start by constructing a polynomial height field representation of the local neighborhood of the point of analysis. We first build an orthonormal *x-y-z* coordinate frame centered at the surface point. At any location (*x,y*), the height (*z*) is a function of *x* and *y:*

$$z(x,y) = \mathrm{J}x^3 + \mathrm{I}y^3 + \mathrm{H}x^2y + \mathrm{G}xy^2 + \mathrm{F}x^2 + \mathrm{E}y^2 + \mathrm{D}xy + \mathrm{C}x + \mathrm{B}y + \mathrm{A}$$

By varying the parameters A through J, we can change the shape of this surface patch. Obviously, parameters J, I, H, and G control the third-order shape, F, E, D control the second-order shape, C and B control the first-order shape, and A controls the zero$^{\text{th}}$-order shape. For simplicity, and without loss of generality, the point of analysis is placed at the origin (0,0) of the polynomial patch.

While the parameters A through J allow us to completely control the shape of the surface patch, they are dependent on the user-defined *x-y* coordinate frame of the patch. Changing the coordinate frame may change A through J, even though the shape may not have changed. As a result, we argue that the parameters A through J are not the purely geometric, intuitive parameters we seek. As was shown in a recent thesis [2], the shape of the local neighborhood of a surface point can be expressed more generically (up to any order) as a sum of angular Fourier components extracted in a cylindrical coordinate system around the *z*-axis.

The second-order behavior is controlled by three parameters. Assuming for the moment that the three parameters A, B, C are all zero, so that the cylinder axis of our Fourier analysis coincides with the surface normal, the three parameters D, E, and F alone will define second-order behavior. Alternatively that same behavior can be described by three Fourier components: a constant term (F2_0) with no angular dependency, which characterizes mean curvature; and a pair of sine and cosine terms with two complete periods around the z-axis that together describe the undulating behavior of a saddle at the point of interest. See the top row of Figure 1 for a graphical depiction, and [2] for the algebraic proof. Typically these two terms are combined into a single Fourier term with maximum amplitude (F2_2) and a phase-shift term that characterizes the direction in which that maximum occurs (one of the principal directions). The principal curvatures then can be expressed as k1 = F2_0 + F2_2 and k2 = F2_0 – F2_2.

We have found a way to capture third-order behavior in a similar manner. In essence, the third-order shape function characterizes the derivative of normal curvature. If all the parameters A through F are set to zero, then the four parameters G, H, I, and J specify pure third-order behavior. Carrying out the same angular Fourier analysis around the *z*-axis as above, we find four relevant components: two with a single period around the *z*-axis, and two more with three periods each. By combining the sine and cosine terms with the same periodicity, we can characterize the whole third-order system (Fig. 1, bottom row) with two amplitudes (F3_1, F3_3) and their corresponding phase shifts (α, β). It is convenient to reference the maximum direction of F3_1 (expressed by α) with respect to a direction specified by the user, and to reference the maximum direction of F3_3 (expressed by β) with respect to the direction of F3_1. See the thesis [2] for the mathematical details.

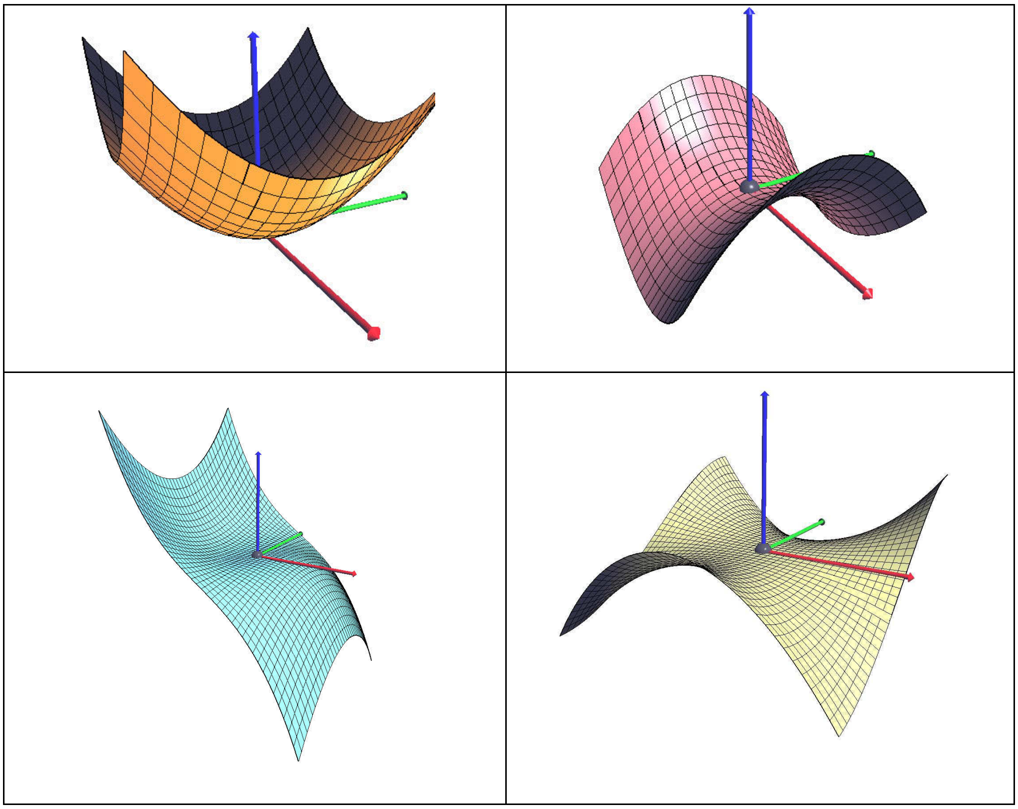

Fig. 1: The top row shows the basis functions for second order shape: F2_0 and F2_2. All second-order shapes are scaled and rotated combinations of these two shapes. Similarly, the bottom row shows the basis functions for third-order shape: F3_1 and, F3_3.

Once this decomposition principle is understood, it is clear how it can be extended to higher order terms, and what we can expect to find there. For the fourth-order characterization we will need five additional parameters: K, L, M, N, O in the Cartesian setting, and they will transform into an additional constant Fourier term (F4_0), another two-period term (F4_2), and a new four-period term (F4_4); the latter two Fourier components are both associated with an angular direction parameter, each of which can again be referenced in the same way as the two direction parameters (α, β). Every next-higher-order term will require one more parameter to fully characterize the new generic undulations that will be possible at that degree of a polynomial surface.

3. USER INTERFACE

A screenshot of the visualization program is shown in Figure 2. The surface patch whose shape we want to study is displayed in the middle, centered at the point of analysis (which is shown as a small sphere). The shape is controlled independently by two sets of sliders. The sliders on the LEFT control the polynomial parameters A through J and thus specify the height of the surface patch above the rectangular domain in a Cartesian coordinate system. The sliders on the RIGHT control the amplitude of the Fourier components and their phase-shifts around the z-axis. The sliders on the right have four parameters (F3_1, α, F3_3, β) to control the third-order shape, three parameters (F2_0, F2_2, γ) to control the second-order shape, and two parameters (F1_1, δ) to control the first-order shape. (As explained in Section 4, γ and δ denote the phase-shifts of the second- and first-order Fourier components.)

The shape of the surface patch can be controlled by either set of parameters. As we change one set of parameters, the sliders corresponding to the other set change simultaneously. In this way, we can readily see the relationship between the two different representations.

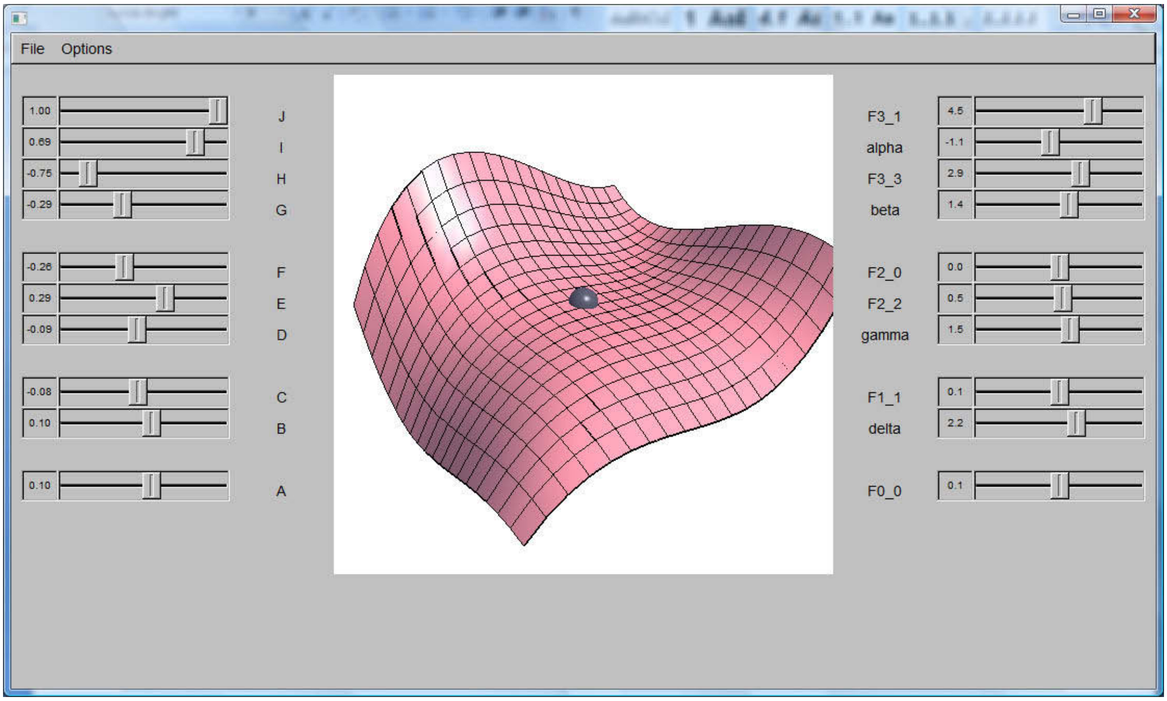

Fig. 2: The user interface for our visualization program. The sliders on the LEFT describe a polynomial height field over a rectangular domain in a Cartesian coordinate system. The sliders on the RIGHT describe the same patch in terms of angular Fourier components around the z-axis. The two sets of parameter values are coupled, so that they both always describe the same geometry, regardless of which slider bank gets manipulated.

In addition to the sliders on each side of the surface patch, our visualization program offers a number of optional display features. For example, using keyboard hotkeys or menu item selections, we can toggle the display of:

1. <u>The second-order surface:</u> With the second-order surface overlaid on the third-order surface, we can spot the differences between the two surfaces and understand the types of shape changes possible by manipulating the third-order shape parameters. See Figure 3.

2. <u>Fourier frame directions:</u> Arrows corresponding to the directions of the peaks and troughs of the Fourier shape components are useful for understanding the orientation of the Fourier shape components and also point out the effects of changing phase-shift shape parameters like α, β. See Figure 4.

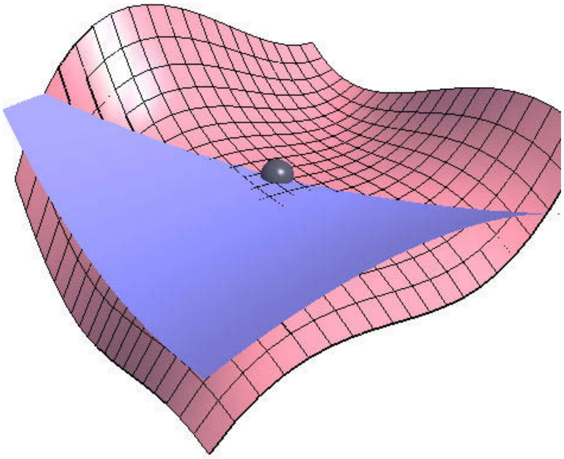

Fig. 3: The quadratic surface at the point of analysis (plain blue) is shown simultaneously with the cubic surface (ruled red) to highlight the difference between the second-order and third-order shape and the extent with which the third-order parameters can affect the surface shape.

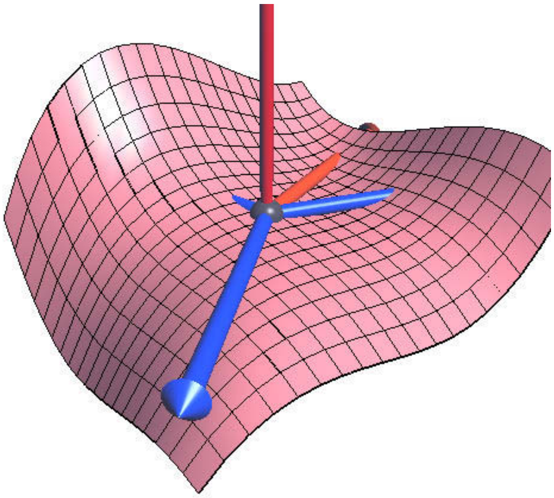

Fig. 4: The orange and blue (horizontal) arrows indicate, respectively, the directions corresponding to the maxima of the first (F3_1) and third (F3_3) Fourier component of third-order shape. The red (vertical) line indicates the direction of the Z axis (that measures height).

4. IMPLEMENTATION

The surface patch is rendered using OpenGl® and the user interface widgets (sliders, menu items, keyboard shortcuts) are built using the "Fast Light Toolkit (FLTK)." The shape of the surface patch is stored by the parameters A through J. Whenever a user changes any of the parameters from one set, we update the values of the other set of sliders. We convert the height equation into polar coordinates:

$$\begin{aligned} z(r, \theta) = r^3 [\, J \cos^3\theta + I \sin^3\theta + H \cos^2\theta \sin\theta + G \cos\theta \sin^2\theta \,] &+ \\ r^2 [\, F \cos^2\theta + E \sin^2\theta + D \cos\theta \sin\theta \,] &+ \\ r [\, C \cos\theta + B \sin\theta \,] + A \end{aligned}$$

We can also express the height as a function of the amplitudes and phase shifts of the Fourier components:

$$z(r,\theta) = r^3 [\ F3\_1\ cos(\theta + \alpha) + F3\_3\ cos3(\theta + \alpha + \beta)\ ] + \\ r^2 [\ F2\_0 + F2\_2\ cos2(\theta + \gamma)\ ] + \\ r\ [\ F1\_1\ cos(\theta + \delta)\ ] + F0\_0$$

By expanding trigonometric identities and grouping and comparing coefficients, we can obtain the Fourier components as a function of the parameters A through J and vice versa. In the related thesis [2], we provide explicit formulae for converting between the third-order parameters (J, I, H, G to F3_1, F3_3, α, β and vice versa).

5. CONCLUSIONS

Using our visualization program, we were able to visualize the exact basis functions that compose surface shapes. Since the basis functions are independent of the surface parameterization and are purely geometric in nature, visualizing their combinations allowed us to develop a strong intuition for the types of shapes that can be formed at a given order.

The question of how to best characterize higher-order surface behavior was raised when we were trying to understand how many different, linearly independent functionals that measure curvature variation could be formulated. The first functional with curvature variation terms was introduced by Moreton and Séquin [6] and focused on the inline derivative of normal curvature. Mehlum and Tarrou [5] and Joshi and Séquin [3] introduced two other functionals that considered curvature derivatives in different directions. Gravesen [1] formulated 18 functionals that contain third-order terms. It was unclear which of these different functionals were contained in the minimal set of basis functionals, and which were simply combinations of the functionals from that minimal set.

In the thesis [2] we show how a number of popular third-order functionals simplify down to terms involving F3_1 and F3_3. In fact, we show that based on our Fourier shape parameters, we can formulate a *complete* set of basis functional components, which can then be combined linearly to form all other third-order functionals. We also show that the functional proposed by Mehlum and Tarrou [5] is a "complete" third-order functional in the sense that other functionals that measure third-order behavior produce information that is measured also by the Mehlum-Tarrou functional.

6. REFERENCES

[1] Gravesen, J.; Ungstrup, M.: Constructing invariant fairness measures for surfaces, Advances in Computational Mathematics, 17(1), 2001, 67–88.

[2] Joshi, P.: Minimizing curvature variation for aesthetic surface design. Ph.D. thesis, Oct 2008. EECS Department, University of California, Berkeley. http://www.eecs.berkeley.edu/Pubs/TechRpts /2008/EECS-2008-129.html

[3] Joshi, P.; Séquin, C. H.: Energy minimizers for curvature-based surface functionals, Computer-Aided Design and Applications, 4(5), 2007, 607–617

[4] Maekawa, T.; Wolter, F.; Patrikalakis, N.: Umbilics and lines of curvature for shape interrogation, Computer-Aided Geometric Design, 13(2), 1996, 133–161.

[5] Mehlum, E.; Tarrou, C.: Invariant smoothness measures for surfaces, Advances in Computational Mathematics, 8(1), 1998, 49–63.

[6] Moreton, H.; Séquin, C. H.: Functional optimization for fair surface design, In: ACM SIGGRAPH 1992, 167–176