CORE
Metadata, citation and similar papers at core.ac.uk
Provided by Bradford Scholars

Ugail H and Kirmani S (2006): "Shape Reconstruction using Partial Differential Equations" ***WSEAS Transactions on Computers*****, 10 (5): 2156-2161.**

Shape Reconstruction using Partial Differential Equations

HASSAN UGAIL and SYED KIRMANI

School of Informatics
University of Bradford
Bradford, BD7 1DP, UK
(h.ugail, s.k.n.kirmani)@bradford.ac.uk

Abstract: - We present an efficient method for reconstructing complex geometry using an elliptic Partial Differential Equation (PDE) formulation. The integral part of this work is the use of three-dimensional curves within the physical space which act as boundary conditions to solve the PDE. The chosen PDE is solved explicitly for a given general set of curves representing the original shape and thus making the method very efficient. In order to improve the quality of results for shape representation we utilize an automatic parameterization scheme on the chosen curves. With this formulation we discuss our methodology for shape representation using a series of practical examples.

Key-Words: - Shape representation, Partial differential equations, Surface generation.

1 Introduction

Representation of a surface implies reconstructing an existing geometry which reflects the vital features of the geometry model. The representation of complex surfaces has been one of the main fields of computer graphics and geometric modeling and over the years many different approaches to this problem have been developed. In an ideal world all shapes would be represented exactly, however the real world usually has more complexity than we are able to represent in geometric modeling, and therefore we approximate. As a rule, the more complex a model is, the more it is approximated. A good representation returns an approximation of the model minimizing the error term.

Now a days there exist many commercial Computer Aided Design systems which employ the conventional polynomial surface modeling schemes to represent the existing geometry. One of them is spline based schemes now dominated by Non Uniform Rational B-Splines (NURBS) [8,9], and is a fine example of the polynomial based surface modeling. There has been a considerable amount work undertaken in the area of surface representation using NURBS.

An example of surface representation using spline based methods has been developed by Piegl [15, 16]. In that it is described how the geometry of a brush handle can be re-created whereby the aim was to produce a smooth surface given a series of cross sectional curves. For this problem, the points per cross section were between 10 to 48 and the number of control points after merging all the patches was nearly 4000. In addition to this, the associated weights also need to be taken into account. In a similar fashion Pottmann [14] describes the approximation of a ruled cylindrical surface using NURBS. Here a surface of bi-degree with 7×25 control points was utilzed.

One of the main drawbacks while using spline based methods is the need for extra storage to define traditional shapes. In some applications of NURBS the combinations of weights can be zero resulting zero denominators and it is a difficult condition to impose during curve and surface fitting. Furthermore, handling possible errors can complicate any NURBS-based geometry program.

Triangular meshes or subdivision schemes [4,11, 6] for surface representation has recently been popular as an alternative to spline based techniques. For example, Hubeli and Gross [10] describe a geometric surface in which Doo-Sabin subdivision scheme is applied on a two manifold surface to represent its geometry. In order to reconstruct the geometry of the shape they have utilized around fifty smoothing steps using the subdivision method. Similarly, Catmull [3] reconstruct a teapot using subdivision algorithm. Although being much more flexible than spline based techniques, triangle meshes also have some restrictions and disadvantages. For example, when applying an extreme deformation to a triangle mesh, certain triangles exhibit strong stretching which leads numerically and visually undesirable triangles that have to be overcome.

Surfaces based on PDEs have recently emerged as a powerful tool for geometric shape modeling [1,7,17,19, 13]. Using this methodology, a surface is generated as the solution to an elliptic Partial Differential Equation (PDE) using a set of boundary conditions. The PDE method is efficient in the sense that it can represent complex three-dimensional geometries in terms of a relatively small set of design variables.

In this paper we discuss how a geometric model can be represented mathematically in order to define the shape as close as possible to the real surface in question. Thus, we show how we can represent an existing geometry of an object as accurately as possible with minimal shape data information.

Thus, in this work we use the fourth order elliptic PDE with four position curves as boundary conditions in order to generate the surface. Furthermore, in addition to the curves two surface parameters associated with the boundary conditions are identified which can be adjusted to provide control over the resulting shape.

2 PDE Surfaces

A PDE surface is a parametric surface patch $\underline{X}(u,v)$, defined as a function of two parameters u and v on a finite domain $\Omega \subset R^2$, by specifying boundary data around the edge region of $\partial\Omega$. Typically the boundary data are specified in the form of $\underline{X}(u,v)$ and a number of its derivatives on $\partial\Omega$. Here one should note that the coordinate of a point u and v is mapped from that point in Ω to a point in the physical space. To satisfy these requirements the surface $\underline{X}(u,v)$ is regarded as a solution of a PDE of the form,

$$\left(\frac{\partial}{\partial u^2} + a^2 \frac{\partial}{\partial v^2}\right)^2 \underline{X}(u,v) = 0, \qquad (1)$$

The PDE given in Equation (1) is of fourth order. Therefore, in order to solve the Equation, four boundary conditions are required. Here four positional curves are taken as the four boundary conditions. These four curves are selected from the original geometry model.

Let Ω be a finite domain defined as $\{\Omega : 0 \le u \le 1, 0 \le v \le 2\pi\}$ such that,

$$\begin{aligned} \underline{X}(0,v) &= \underline{P}_0(v), \\ \underline{X}(s,v) &= \underline{P}_s(v), \\ \underline{X}(t,v) &= \underline{P}_t(v), \\ \underline{X}(1,v) &= \underline{P}_1(v), \end{aligned} \qquad (2)$$

where $\underline{P}_0(v)$ and $\underline{P}_1(v)$ define the edges of the surface at $u = 0$ and $u = 1$ respectively. $\underline{P}_s(v)$ and $\underline{P}_t(v)$ are the positions of the second and third curves respectively as shown in the Figure 1(a). Here s and t are the positions of the interior curves such that, $0 \le s < t$, and $s < t \le 1$.

Using the method of separation of the variables, the explicit solution of Equation (1) can be written as,

$$\begin{aligned} \underline{X}(u,v) &= \underline{A}_0(u) \\ &+ \sum_{n=1}^{\infty}\{\underline{A}_n(u)\cos(nv) + \underline{B}_n(u)\sin(nv)\} \end{aligned} \qquad (3)$$

where,

$$\underline{A}_0 = \underline{a}_{00} + \underline{a}_{01}u + \underline{a}_{02}u^2 + \underline{a}_{03}u^3, \qquad (4)$$

$$\begin{aligned} \underline{A}_n(u) &= \underline{a}_{n1}e^{anu} + \underline{a}_{n2}ue^{anu} \\ &+ \underline{a}_{n3}e^{-anu} + \underline{a}_{n4}ue^{-anu} \end{aligned} \qquad (5)$$

$$\begin{aligned} \underline{B}_n(u) &= \underline{b}_{n1}e^{anu} + \underline{b}_{n2}ue^{anu} \\ &+ \underline{b}_{n3}e^{-anu} + \underline{b}_{n4}ue^{-anu}, \end{aligned} \qquad (6)$$

where $\underline{a}_{00}, \underline{a}_{01}, \underline{a}_{02}, \underline{a}_{03}, \underline{a}_{n1}, \underline{a}_{n2}, \underline{a}_{n3}, \underline{a}_{n4}, \underline{b}_{n1}, \underline{b}_{n2}, \underline{b}_{n3}$ and $\underline{b}_{n4}$ are vector valued constants, whose values are determined by the imposed boundary conditions at $u = 0, u = s, u = t$ and $u = 1$.

For a given set of boundary conditions, in order to define the various constants in the solution, it is necessary to perform Fourier analysis of the boundary conditions and identify the various Fourier coefficients. For a finite number of Fourier modes N (say $4 \le N \le 10$) the approximate surface solution can be defined as,

$$\begin{aligned} \underline{X}(u,v) &= \underline{A}_0(u) \\ &+ \sum_{n=1}^{N}\{\underline{A}_n(u)\cos(nv) + \underline{B}_n(u)\sin(nv)\} + \underline{R}(u,v) \end{aligned} \qquad (7)$$

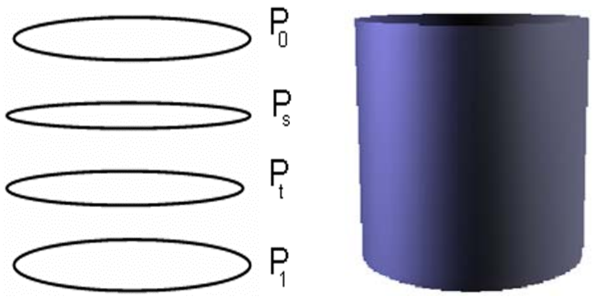
where $\underline{R}(u,v)$ is called a remainder function defined as,

$$\underline{R}(u,v) = \underline{r}_1(v)e^{wu} + \underline{r}_2(v)ue^{wu} + \underline{r}_3(v)e^{-wu} + \underline{r}_4(v)ue^{-wu}, \tag{8}$$

where $\underline{r}_1, \underline{r}_2, \underline{r}_3, \underline{r}_4$ and w are obtained by considering the difference between the original boundary conditions and the boundary conditions satisfied by the function,

$$\underline{F}(u,v) = \underline{A}_0(u) + \sum_{n=1}^{N} \{\underline{A}_n(u)\cos(nv) + \underline{B}_n(u)\sin(nv)\}. \tag{9}$$

The remainder function $\underline{R}(u,v)$ is calculated by means of the difference between the original boundary conditions and the boundary conditions satisfied by the function $\underline{F}(u,v)$ therefore it guarantees that the chosen boundary conditions are exactly satisfy [5].



(a) **(b)**

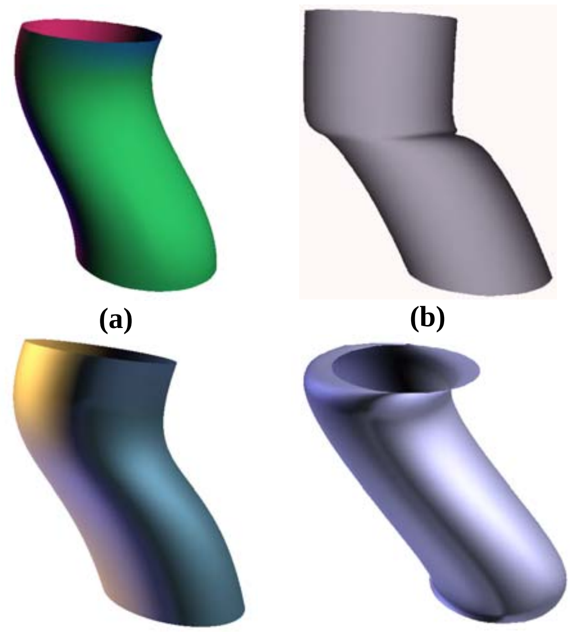
Figure 1. Surface representation using fourth order PDE. **(a)** Four curves representing the conditions necessary for the 4th order PDE. **(b)** The resulting surface.

Figure 1 shows a typical example of the PDE surface representation scheme discussed above. First a set of four curves in the physical space are chosen as shown in Figure 1(a). These four curves are utilized as the conditions to solve the PDE Equation (1) where the resulting shape is shown in Figure 1(b). Note that the resulting shape is a smooth interpolation of the chosen four curves whereby the shape can be controlled solely by the curves.

2.1 Choice of Parameters

Earlier we have introduced two parameters "s" and "t" defined over the parametric domain (u,v) as described in Equation (2). The choice of s and t representing the relative positions of interior curves plays an important role in determining the shape of the resulting surface. It is important to note that since the parameter "u" is ranging from 0 to 1, the values of parameters s and t will lie between 0 and 1 too. Here we show how these two parameters affect the overall geometry of the resulting surface.

Consider the shapes shown in Figure 2. Here the shape shown in Figure 2(a) is the original shape which we wish to reconstruct using the methodology described previously. The shapes shown in Figures 2(b), 2(c) and 2(d) are shapes which have been produced using different values for the parameters s and t.



(a) **(b)** **(c)** **(d)**

Figure 2. Effect of geometry shape for different values of the parameters s and t.

Table 1 summarises the various values of s and t used. The table also shows the error terms involved in approximating the original surface shape where the difference between the original surface (denoted by X_i) and that generated using different values s and t (denoted by $X_i^{'}$) are also shown.

Figure	s	t	$\sum(X_i - X_i')^2$.
3(b)	0.4	0.6	0.8055
3(c)	0.3	0.7	0.0723
3(d)	0..2	0.8	0.9291

Table 1. Values of s and t used in generating the shapes shown in Figure 2.

Both from Figure 2 and Table 1 one can see that the shape shown in Figure 3(c) is the best approximation.

From the above discussions it is notable that the values of s and t have dramatic effects on the overall shape of the resulting surface and hence the representation of the original geometry. Thus, it becomes important to choose these parameters in a proper fashion so as the representation of the original geometry is described with minimal possible error.

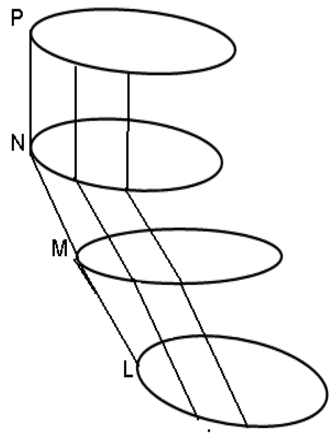


Figure 3. Description of how the parameters s and t are automatically chosen.

Here we describe a technique that is used to automatically define the values of the parameters which we call as *flexible parameterization* or point by point calculation of the parameters.

Consider Figure 3 which illustrates how the parameters s and t are automatically chosen. Let L, M, N and P be the initial points on the four boundary curves relative to $v = 0$ respectively. Then we define,

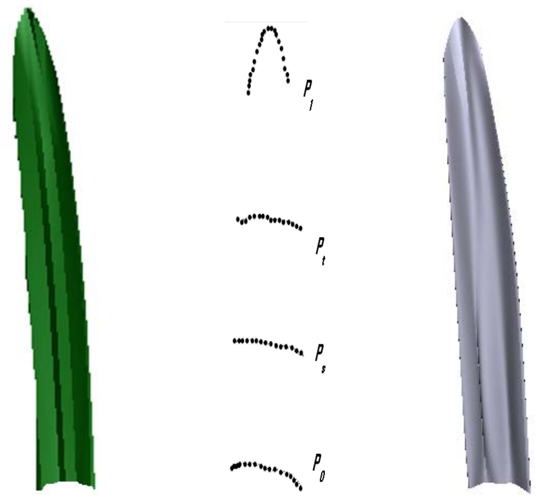
$$s = \frac{LM}{LM + MN + NP} \ , \tag{10}$$

$$t = \frac{LM + MN}{LM + MN + NP} \, . \tag{11}$$

Here LM, MN & NP are the Euclidian distances among the points of boundary curves. Hence for all corresponding points on four boundary curves there are different values of the parameters s and t.

3 Results

In this section we show the results from surface representations we have performed using the methodology described above. In particular, we show three examples; first the reconstruction of a leaf shape of a plant; second the reconstruction of a plant shape and finally the reconstruction of a human face where the original data were available from a 3D scanner.



(a) **(b)** **(c)**

Figure 4. The reconstruction of the shape of a leaf. **(a)** The original shape **(b)** Four curves extracted from the geometry of leaf. **(c)** The reconstructed shape.

As a first example for demonstrating the technique proposed here for shape reconstruction, we take the example of the leaf shown in Figure 4. Figure 4(a) shows the original shape of the leaf. From this shape we then extracted four curves which are shown in Figure 4(b). Looking at the curves we can see that the curve P_1 is not of the same shape as that of the remaining curves. It means the distances between points on the curves P_1 and curve P_t are not the same. Therefore, the flexible paramterization for the parameters s and t are adopted. Using this techniques we then solve the PDE which results in a surface as shown in Figure 4(c). As one can see that the two shapes (Figure 4(a) and Figure (c)) agree closely. The difference between the two shapes was also numerically tested by means of comparing the two surfaces. Thus, if we denote X_i representing the points on the original surface and X_i' to be the points on the reconstructed surface then the error term $\sum (X_i - X_i')^2 = 0.0117$. Hence it shows there is good agreement between the two shapes.

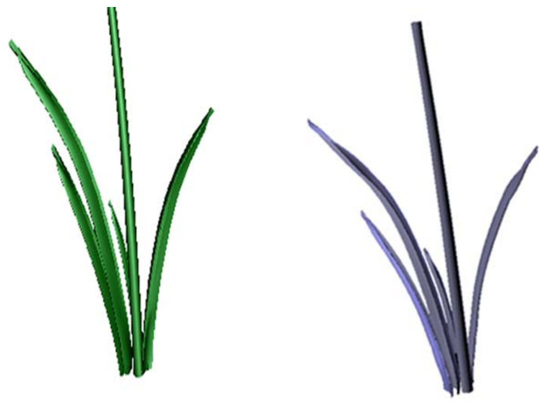


Figure 5. The reconstruction of the shape of a plant. **(a)** The original shape **(b)** The reconstructed shape.

As a second example, we discuss the reconstruction of the shape of a plant as shown in Figure 5. Figure 5(a) shows the original shape of a plant. Similar to the previous example, from the shape of the original plant, we extracted a series of curves. These curves were then utilized to solve the PDE. The resulting shape of the plant is shown in Figure 5(b). As one can see that there is close agreement between the two shapes.

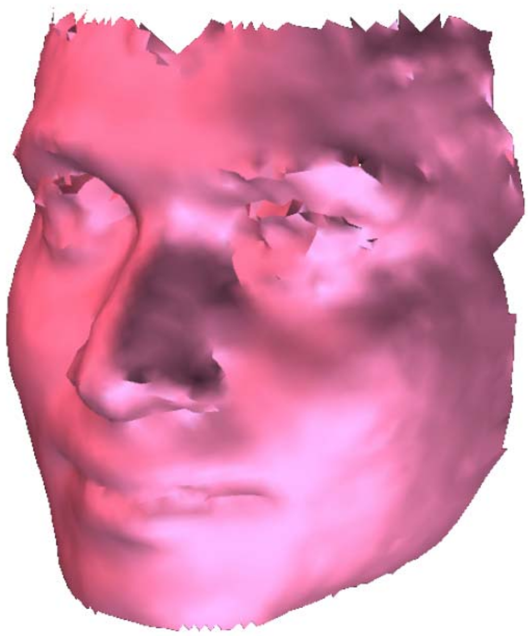


Figure 6. Scan data corresponding to a 3D face.

As a final example to demonstrate the techniques utilized here we reconstruct the facial geometry where the original data, as shown in Figure 6, were available through a 3D scanner. From this original data we automatically extracted a series of curves. These curves were then utilized to reconstruct the shape of the face. Figure 7 shows the result which represents the smooth surface generated. As can be seen clearly there is good visual correspondence between the original facial data and that generated.

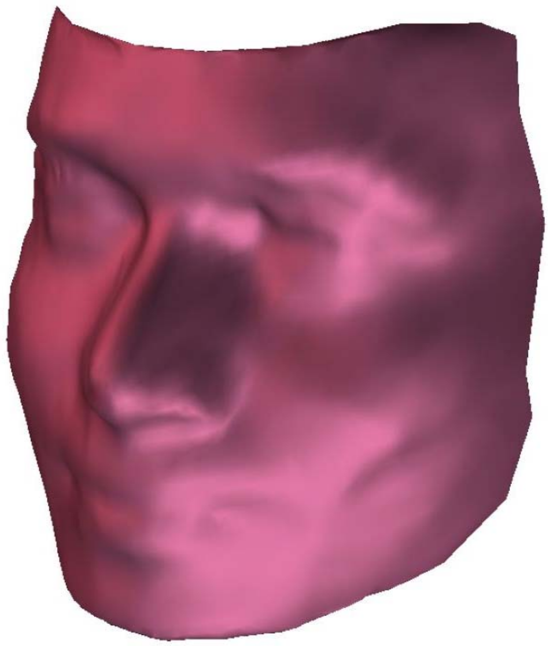


Figure 7. Reconstructed 3D face.

4 Conclusion

In this paper we describe a technique for reconstructing surfaces. We utilize a boundary value approach whereby the solution of an elliptic PDE is solved for appropriately defined curves in the physical space corresponding to the original geometry in question. The chosen PDE is solved explicitly which enables efficient computation of the surfaces. Furthermore, due to the availability of explicit solution we can undertake arbitrary level of surface refinement, once an initial reconstruction has been performed.

It is also noteworthy to point out that similar surface generation schemes already exist in the literature. For example, Bloor and Wilson [1] originally proposed the method for surface generation based on the Biharmonic equation. Their original work as well as the subsequent work [2,12,17,18] included solving the Biharmonic equation in a classical way whereby the usual function and derivative boundary conditions are taken at the edges of the surface patch. Thus, in these earlier schemes only the edge boundary curves are satisfied whilst the interior curves are utilised for defining the normal boundary condition at the edges of the surface patch. Detailed discussions on this method can be found in [1,17, 18].

Apart from the good reconstruction results we achieve through the proposed method there are several advantages to our proposed scheme which has an analytic representation of the shape we are reconstructing. These include efficient visualisation, efficient rendering of the facial data, intuitive manipulation of the data and efficient computation of surface quantities such as curvature.

Acknowledgements

H. Ugail wishes to acknowledge the support received from UK Engineering and Physical Sciences Council grant EP/C015118/01 and EP/D000017/01 through which this work was completed.

References:

[1] M.I.G. Bloor and M.J. Wilson, Generating Blend Surfaces using Partial Differential Equations, *Computer Aided Design*, Vol. 21, No. 3, 1989, pp 165-171.

[2] M.I.G. Bloor and M.J. Wilson, The Efficient Parameterization of Generic Aircraft Geometry, *Journal of Aircraft*, Vol. 32, No 6, 1995, pp.1269-1275.

[3] E. Catmull, *Subdivision Algorithm for the Display of Curved Surfaces*, PhD thesis, University Of Utah, 1974.

[4] E. Catmull and J. Clark, Recursively Generated B-Spline Surfaces on Arbitrary Topological Meshes. *Computer Aided Design*, Vol. 10, No. 6, 1978, pp. 350-355.

[5] C. W. Dekanski, M.I.G. Bloor and M. J.Wilson, The Representation of Marine Propeller Blades using the PDE Method, *Journal of Ship Research*, Vol. 38, No. 2, 1995, pp. 108-116.

[6] N. Dyn, D. Levine and J. A. Gregory, A Butterfly Subdivision Scheme for Surface Interpolation with Tension Control, *ACM Transactions on Graphics*, Vol. 9, No. 2, 1990, pp. 160-169.

[7] H. Du and H. Qin, A Shape Design System using Volumetric Implicit PDEs, *Computer Aided Design*, Vol. 36, No. 11, 2004, pp.1101-1116.

[8] I. D. Faux and M. J. Pratt, *Computational Geometry For Design And Manufacture*, Halsted Press (Wiley), 1979.

[9] G. Farin, From conics to NURBS, A Tutorial and Survey, *IEEE Computer Graphics and Applications*, Vol. 12, No. 5, 1992, pp. 78-86.

[10] A. Hubeli, and M. Gross, *A survey of Surface Representations for Geometric Modeling*, Computer science department, ETH Zurich, Swizerland, CS Technical report No. 335, 2000.

[11] C. T. Loop. *Smooth Subdivision Surfaces based on Triangles*, Master's Thesis, Department of Mathematics, University of Utah, 1987.

[12] T.W. Lowe, M.I.G. Bloor, and M. J. Wilson. Functionality in Blend Design, *Advanced in Design Automation, Vol 1, Computer Aided and Computational Design ASME*, New York, 1990, pp. 43-50.

[13] J. Monterde and H. Ugail, A General 4th-Order PDE Method to Generate Bézier Surfaces from the Boundary, *Computer Aided Geometric Design*, Vol. 23, No. 2, 2006, 208-225.

[14] H. Pottmann, S. Leopoldseder, M. Hofer, T. Steiner and W. Wang, Industrial Geometry: Recent Advances and Applications in CAD, *Computer Aided Design*, Vol. 37, No. 7, 2005, pp. 751-766.

[15] L. Piegl, Recursive Algorithms for the Representation of Parametric Curves and Surfaces, *Computer Aided Design*, Vol. 17, No. 5, 1985, pp. 225-229.

[16] L. Piegl, *The NURBS Book*, Springer-Verlag, New York, 1997.

[17] H. Ugail, M.I.G. Bloor, and M.J. Wilson, Techniques for Interactive Design Using the PDE Method, *ACM Transactions on Graphics*, Vol. 18, No. 2, 1999, pp. 195-212.

[18] H. Ugail, Spine Based Shape Parameterisations for PDE Surfaces, *Computing*, Vol. 72, 2004, pp. 195-206.

[19] L. You, P. Comninos, J.J. Zhang, PDE Blending Surfaces with C2 Continuity, *Computers and Graphics*, Vol. 28, No. 6, 2004, pp.895-906.