Surface fitting based on a feature sensitive parametrization

Yu-Kun Lai [a,*] Shi-Min Hu [a] Helmut Pottmann [b]

[a] *Tsinghua University, Beijing, China*

[b] *Vienna Univ. of Technology, Wiedner Hauptstr. 8–10/104, 1040 Wien, Austria*

Abstract

Most approaches to least squares fitting of a B-spline surface to measurement data require a parametrization of the data point set and the choice of suitable knot vectors. We propose to use uniform knots in connection with a feature sensitive parametrization. This parametrization allocates more parameter space to highly curved feature regions and thus automatically provides more control points where they are needed.

Key words: surface approximation, parametrization, feature sensitivity

1 Introduction

In data fitting with B-spline surfaces, both parametrization and the choice of the knot vectors are difficult and also closely related problems [23]. The number of knot lines in some part of the parameter domain is in direct relation to the number of control points in the corresponding part of the surface. Moreover, more control points are needed in *feature regions* such as sharp edges, smoothed edges, ridges, valleys and prongs.

The present short paper presents a solution to this problem by suggesting to use a *feature sensitive (fs) parametrization* for surface fitting. A uniform choice of knots over a parameter domain which results from a fs parametrization automatically provides more control points for feature areas, since it allocates

* Corresponding author.

Email addresses: laiyk@cg.cs.tsinghua.edu.cn (Yu-Kun Lai), shimin@tsinghua.edu.cn (Shi-Min Hu), pottmann@geometrie.tuwien.ac.at (Helmut Pottmann).

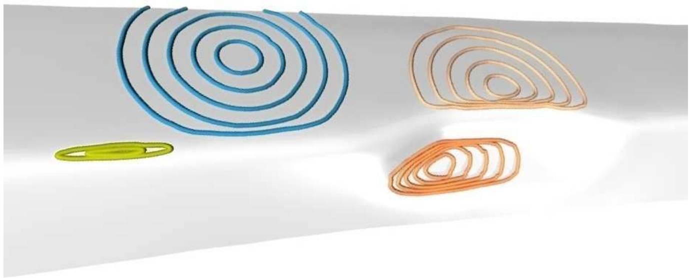

Fig. 1. Isolines of the distance from given points computed with respect to the feature sensitive metric.

more parameter space for feature regions. We will show how to compute such a fs parametrization and illustrate its effect at hand of examples.

1.1 Previous work

Since parametrization and the choice of the knots are essential for most B-spline curve and surface fitting methods, there is a relatively large body of literature on it. For curve parametrization and knot placement methods, we refer to [9,14,13]. The state of the art on surface approximation from the CAD perspective is found in [25]. Let us also mention that there are fitting techniques which do not require a parametrization [17,16]; they need, however, an initial guess for the optimization, which may be obtained with the methods presented in this paper (for an example, see Section 3).

Parametrization is not only important for least squares fitting. It is a key step in a number of geometry processing techniques and thus received a lot of attention in recent years. For a survey, we refer to [7]. For many applications, such a parametrization should be near-isometric (exact isometry being achievable only for developable surfaces). Practical parametrization methods may achieve conformality (angle-preservation), area-preservation or a tradeoff between those two [4].

Since the present paper deals with a feature sensitive method, we also give a few references on feature sensitive geometry processing. Feature extraction is either performed by estimating differential quantities via local or global surface fitting (see [15] and the references therein) or based on appropriate integral invariants such as moments of local neighborhoods [3]. Feature sensitivity mostly has been investigated in connection with specific applications, e.g., fs surface extraction from volume data [12], fs sampling for remeshing [2], fs remeshing based on curvature estimation [24,1], fs geometry images [21,22], fs piecewise planar approximation [5] or a PDE approach to fs surface editing [3]. For fitting of measurement data, work on fs filtering and smoothing [8,10] is certainly of interest.

2 The feature sensitive metric

Our approach is based on a feature sensitive metric which has so far been used for fs morphology on surfaces [18] and for the design of curves on surfaces which are well aligned with the surface features [16].

Roughly speaking, features are characterized by the way in which the unit surface normal varies along the surface Φ. It is therefore natural to consider the field of unit normal vectors $\mathbf{n}(\mathbf{x})$ attached to the surface points $\mathbf{x} \in \Phi$ as a vector-valued image defined on the surface. Borrowing the idea of an image manifold from Image Processing [11], one can now map each surface point $\mathbf{x}$ to a point $\mathbf{x}_f = (\mathbf{x}, w\mathbf{n})$ in $\mathbb{R}^6$. Here, w denotes a non-negative constant, whose magnitude regulates the amount of feature sensitivity and the scale on which one wants to respect features (see Section 2.1). In this way, Φ is associated with a 2-dimensional surface $\Phi_f \subset \mathbb{R}^6$. By measuring distances of points and lengths of curves on Φ_f instead of Φ, we introduce a feature-sensitive metric on the surface [18]. As shown in Fig. 1, distances across features are much larger in the fs metric than with respect to the ordinary Euclidean one.

The key for our application is the computation of a parametrization of a surface Φ (which may be a triangulated set of measurement points) with help of a parametrization of its image manifold Φ_f. Thus, in the remainder of this section we deal with the computation of Φ_f.

We would like to point out that the use of $\Phi_f \subset \mathbb{R}^6$ is mainly for a simple introduction of the fs metric. As will be seen from the developments given below, we can still explain everything in $\mathbb{R}^3$ via an appropriately combined processing of points and normals. The geometry of the image manifold in $\mathbb{R}^6$ tells us how to combine point and normal information, but it does not result in any computational overhead over working in 3D.

2.1 Computation of the image manifold

The computation of the image manifold Φ_f requires surface normals. For a smooth surface in any representation this is a simple task. However, we need to be careful with the following issues: the presence of noise, the scale, and the presence of sharp features. The latter can be *edges* as intersection curves of smooth surfaces or *corners*, which are points, where at least three surface patches intersect or where the local shape is like the vertex of a cone.

Noise and scale. We assume that we are given an error tolerance δ for points on the model and a parameter ε (usually small, but much larger than δ); only features of width $> \varepsilon$ shall be handled.

In the presence of noise or negligible features, we estimate normals from a neighborhood of size $\approx \varepsilon$, e.g., with local planar or quadratic fits (see e.g. [23]) and a fitting error $< \delta$. Even if this does not mean smoothing of the original data, this approach prevents a dramatic increase of the noise level in Φ_f. Moreover, marginal features – in contrast to relevant ones – do not manifest themselves in larger areas of Φ_f.

If the model Φ gets scaled by a factor σ, Φ_f scales with the same factor if the weight w is also multiplied by σ. Hence, w has to be judged in relation to the object size. Suitable values of w for certain purposes will therefore be given under the assumption that the model fits into the unit cube.

Sharp features. In order to carefully represent a sharp feature in a B-spline surface, it must be a parameter line. If this is not the case, the best we can do is to approximate it by a smoothed edge with very high curvature across the edge. Thus, we assume the viewpoint that a sharp feature is a limit case of a smooth surface. The reader may consider sharp features smoothed with a very small blending radius. Then, a point $\mathbf{p}$ on a sharp edge $\mathbf{c} \subset \Phi$, with normals $\mathbf{n}^-$ and $\mathbf{n}^+$ of the adjacent smooth surfaces, corresponds to a circular arc $\mathbf{p}_f$ on the image manifold Φ_f; this arc has the endpoints $(\mathbf{p}, w\mathbf{n}^-)$ and $(\mathbf{p}, w\mathbf{n}^+)$. We have a blow-up phenomenon (see Fig. 2): A sharp edge is mapped to a surface region on Φ. Likewise, at a corner we have a two-dimensional set of surface normals and a corresponding spherical patch in the image manifold. This phenomenon is already known from (untrimmed) offsets at a distance μ, which incidentally can be obtained from Φ_f via the mapping $(x_1, \ldots, x_6) \mapsto (x_1, x_2, x_3) + \frac{\mu}{w}(x_4, x_5, x_6)$.

Because of the wide usability, we focus on surfaces Φ which are given as a *triangle mesh.* After normals have been estimated, we can simply map each vertex to feature space $\mathbb{R}^6$ while keeping the connectivity unchanged. Thus, Φ_f is represented by a triangle mesh embedded in $\mathbb{R}^6$. However, sharp features and corners with large normal changes require a special treatment in order to represent the image manifold with sufficient accuracy.

Detection of Sharp Edges and Corners. A mesh representation generally does not contain explicit information on sharp edges or corners. Thus, at the first stage of the algorithm, we need to identify those features. This can be done as follows: (1) For each edge segment e in the mesh, we compute a robust normal deviation angle ν. For well-shaped adjacent triangles and well-sampled models without data errors, ν is the angle between the normals of the two adjacent faces. In critical cases, we intersect the mesh locally with a plane through e's midpoint $\mathbf{m}$ and orthogonal to e. With robust fits (by a straight line or a low degree polynomial) of the profile section data on either side of $\mathbf{m}$, the normal deviation angle ν is estimated. (2) With a user-defined threshold β, an edge segment e belongs to a sharp edge if $\nu > \beta$. (3) Corners are detected where

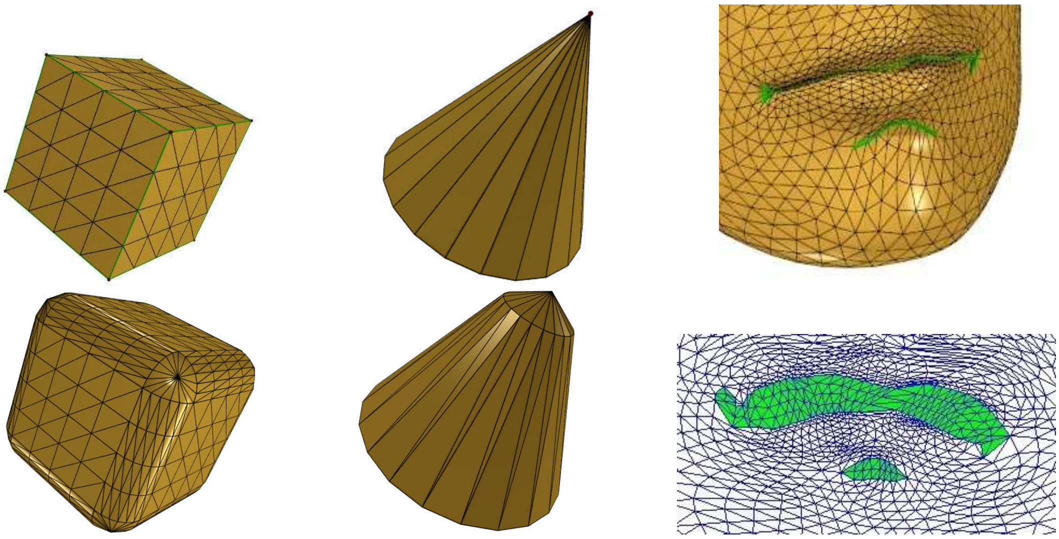

Fig. 2. The blow-up phenomenon at sharp edges and corners: Top: original mesh in $\mathbb{R}^3$. Bottom left and center: projection of the corresponding mesh in $\mathbb{R}^6$. Bottom right: Parametrization of the corresponding mesh in $\mathbb{R}^6$.

three or more sharp edges coincide. A corner $\mathbf{v}$ of the cone-vertex type is found as follows: Let γ_i denote the angles of the adjacent triangles at $\mathbf{v}$, then the vertex $\mathbf{v}$ is seen as a corner if $\sum_i \gamma_i/(2\pi) < \cos(\beta/2)$.

Edge/Corner Handling. In order to handle sharp edges and corners in a consistent way, we consider five classes of vertices. Sharp edge segments form connected paths: an interior vertex of the path is called *in-path vertex*, each end point is a *path-end vertex*. A *boundary vertex* is placed at the boundary of the mesh. A *corner* has been explained above. Any other vertex is an *ordinary vertex*. An ordinary vertex is not blown up, and neither are boundary and path-end vertices. An in-path vertex will be split according to the change in surface normals there. The edges connecting a path-end vertex and an in-path vertex or two in-path vertices will be blown up to a region in $\mathbb{R}^6$ that is triangulated appropriately (see Fig. 2). If a vertex $\mathbf{v}$ is a corner, but its neighbors are not, it is mapped to a submesh C_f in $\mathbb{R}^6$ as follows: An average surface normal at $\mathbf{v}$ yields the center of C_f. An edge emanating from $\mathbf{v}$ yields one or more vertices of C_f depending on whether it is sharp or not. Two adjacent corners (a rare occurrence) are avoided by inserting a further vertex between them.

3 B-spline surface fitting based on a feature sensitive parametrization

Parameterizing a mesh Φ over a planar domain D requires to set up a bijective mapping between Φ and D. This is a key step in a number of geometry processing techniques including surface fitting. For several applications, but not necessarily for surface fitting, such a parametrization should be near-isometric (exact isometry being achievable only for developable surfaces). Practical parametrization methods may achieve conformality (angle-preservation), area-preservation or a tradeoff between those two [4,7]. Let us see what we can achieve by parameterizing Φ via an appropriate area-preserving parametrization of Φ_f: We will see that the resulting *fs parametrization* assigns rather more space of the parameter domain D to highly curved regions than it does to flat ones.

As mentioned, we are especially interested in *area preserving mappings* $\Phi_f \mapsto D$. In order to give a more precise explanation of their effect, we mention the following property whose proof is outlined in the Appendix.

Theorem 1 *Given a region $R \subset \Phi$, the surface area A_f of the corresponding region R_f in the image manifold Φ_f is expressed via the principal curvatures κ_1, κ_2 and Gaussian curvature $K = \kappa_1\kappa_2$ of Φ as*

$$A_f = \int\limits_R \sqrt{1 + w^2(\kappa_1^2 + \kappa_2^2) + w^4K^2}\; dA. \tag{1}$$

Here dA is the area element of Φ.

This has a very useful effect on our parametrization. For large values of w, the surface area A_f is governed by the value of $A_w := w^2 \int |K| dA$. Therefore, the main growth $A_f - A$ in surface area of corresponding regions on Φ_f and Φ happens at places of Φ which have large Gaussian curvature K. We could also say that the overhead in surface area on Φ_f is in a direct relation to the deviation of the corresponding region $R \subset \Phi$ from a developable surface (a surface characterized by $K = 0$). Note that only developable surfaces possess a distortion free (isometric) parametrization over a planar domain D. If Φ is a developable surface, one principal curvature vanishes, say $\kappa_1 = 0$. Since the other principal curvature κ_2 still may exhibit a large variation, it would not be advisable to use an isometric mapping and uniform knots in a parametrization for fitting such a surface. Our method takes this into account: For a developable surface and large w, A_f is governed by $w \int |\kappa_2| dA$. Thus, regions with high κ_2 on Φ will get enlarged on Φ_f. This is precisely what we want to have.

Let us now assume that we have constructed an area preserving parametrization of Φ_f. Such a parametrization is feature sensitive, since it reserves parameter space according to the value of A_f in (1), which is a kind of total curvature of Φ. Highly curved regions get more space than others in a sense discussed above. This effect is also seen in Fig. 3. In Fig. 4, the blow-up effect is visualized with stretch-related color coding. We are talking here about the stretch between the actual model Φ and the image manifold Φ_f. Since the parametrization in Fig. 4 has been computed with a stretch minimizing parametrization of Φ_f, the stretch between Φ and Φ_f can also be observed as stretch between Φ and the parameter domain. Note that the red parts in the figures indicate large stretching, which correspond exactly to the feature regions of the model.

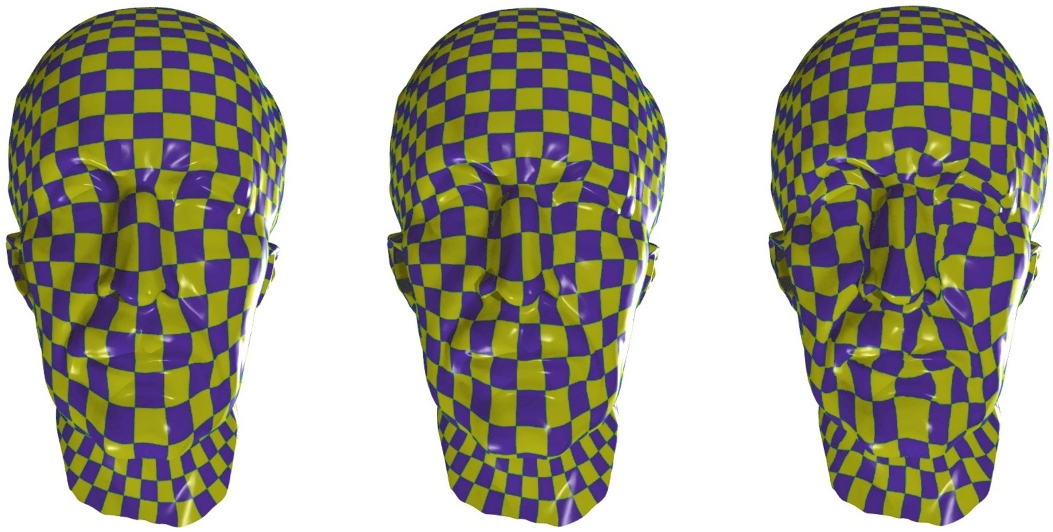

Fig. 3. Stretch minimizing parametrization with increasing feature sensitivity: $w = 0, 0.08, 0.25$

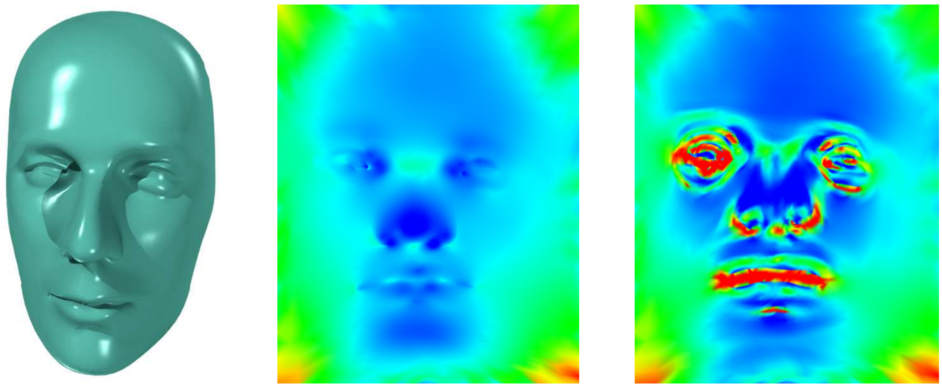

Fig. 4. Visualization of the parameter domain with stretch-related color coding. Left: the model; center: parametrization without feature sensitivity; right: fs parametrization.

Let us briefly describe *parametrization by stretch minimization* [19], since it is heavily used in our work: At first, the boundary of a patch is mapped to a rectangular domain. Since stretch minimization is a nonlinear optimization problem, one requires an initial parametrization, which is set up with a robust and computationally efficient method like mean value parametrization [6]. The texture stretch metric is defined as the root-mean-square stretch over all directions and optimized with iterative local line search optimization. As it is a nonlinear optimization problem, it is slow for large models. Thus, we employ a hierarchical approach as in [20] to increase both efficiency and quality of the parametrization.

In a fs parametrization, sharp features, if handled as those, get blown up (see Fig. 2); then we do not have a parametrization of Φ in the usual sense, but still a practically useful tool, which is shown in the following by means of B-spline surface fitting.

B-spline fitting based on a fs parametrization is illustrated in Fig. 5. We parameterize the model over a rectangular domain with a *fs stretch minimizing parametrization*, that is, a stretch minimizing parametrization [19] of Φ_f. Then we fit the data with a uniform cubic B-spline surface (30×20 control points), based on the standard regularized least squares fitting algorithm [23]. The fs approach is superior at sharp and smooth feature areas. Sharp edges of the model always get smoothed by fitting (unless we have multiple knot lines there, which is only possible in special cases), but the rounding effect is smaller with the fs approach.

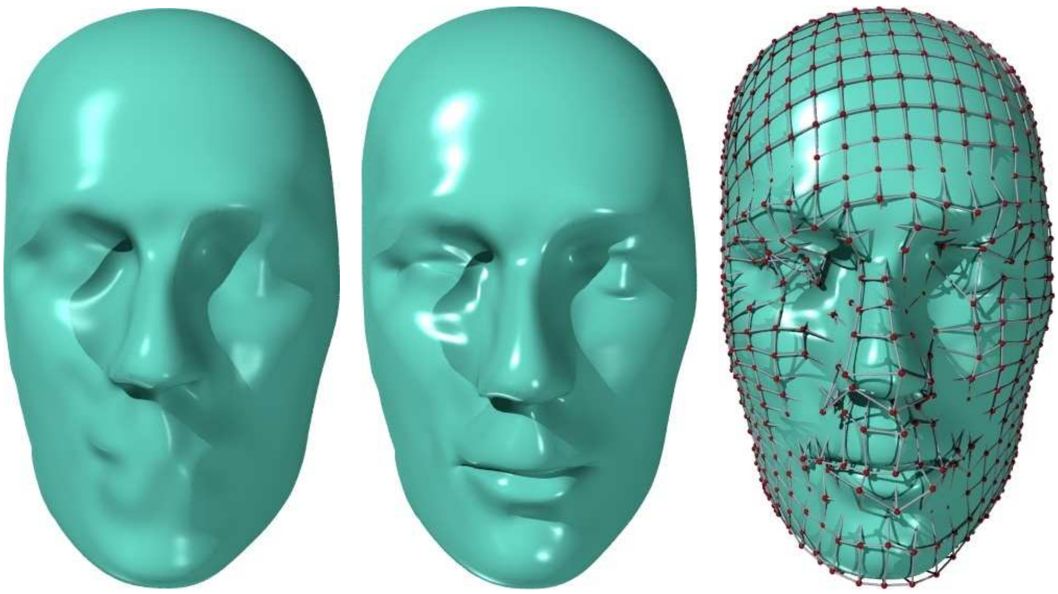

Fig. 5. B-spline fitting. Without (left) and with (center) fs parametrization ($w = 0.07$). Right: control grid.

An example of fitting the screwdriver part with periodic B-spline surfaces is given in Fig. 6, and fitting errors are color coded. The red parts are regions with high fitting error, while the blue parts are those with low error. Each fitting surface contains 30×30 control points. Control grids are illustrated in Fig. 7. Fig. 8 shows the fitting results on a femur model, again using 30×30 control points. Compared to the result without feature sensitivity, the fs approach preserves more significant details.

Our approach provides a good initial parametrization of mesh models suitable

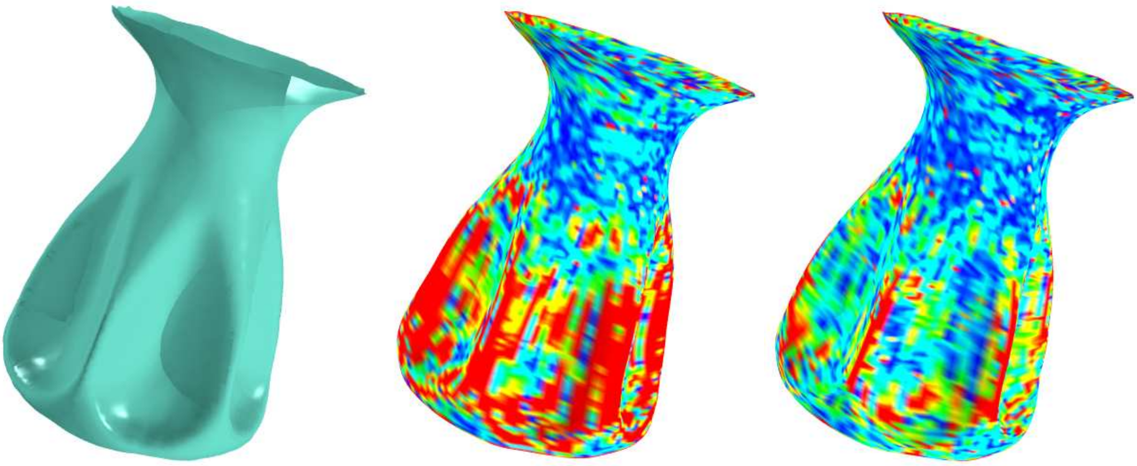

Fig. 6. Fitting of a part on a screwdriver (left) using a periodic B-spline surface without (center) and with (right) feature sensitivity, $w = 0.20$.

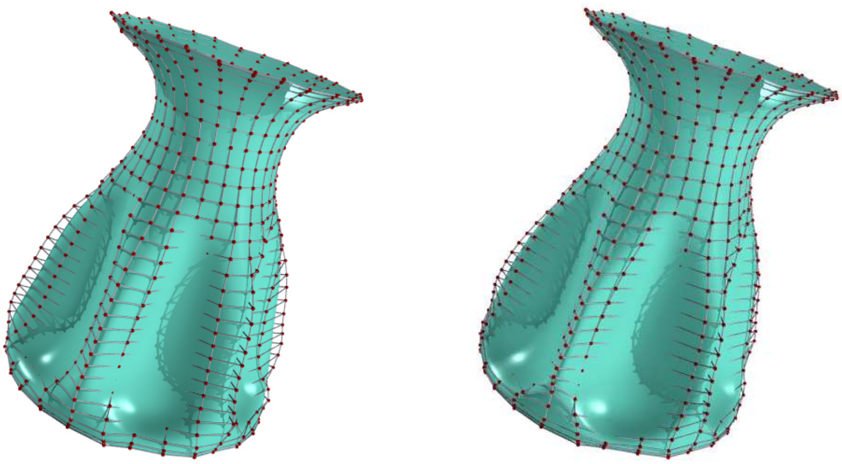

Fig. 7. Control grids of B-spline surfaces of Fig. 6 (middle and right) obtained by fitting without (left) and with (right) feature sensitivity.

for B-spline surface fitting. After least-squares fitting, iterative methods can be used to further improve the result. We tested this with the Newton-type algorithm in [17], which is based on quadratic approximation of the squared distance field and denoted by *SDM* in the following. For all iterative algorithms in nonlinear optimization, good initial positions usually lead to better results or faster convergence. Clearly, also SDM does not make an exception to this rule. Fig. 9 shows the results of SDM optimization. If SDM gets initialized with a fit obtained by a fs parametrization, it converges to a much better result. In Fig. 10, the car part on the left is approximated with a B-spline surface with 20×30 control points using a fs parametrization (center) and SDM is then successfully used to improve that result; corresponding control grids are shown in Fig. 11.

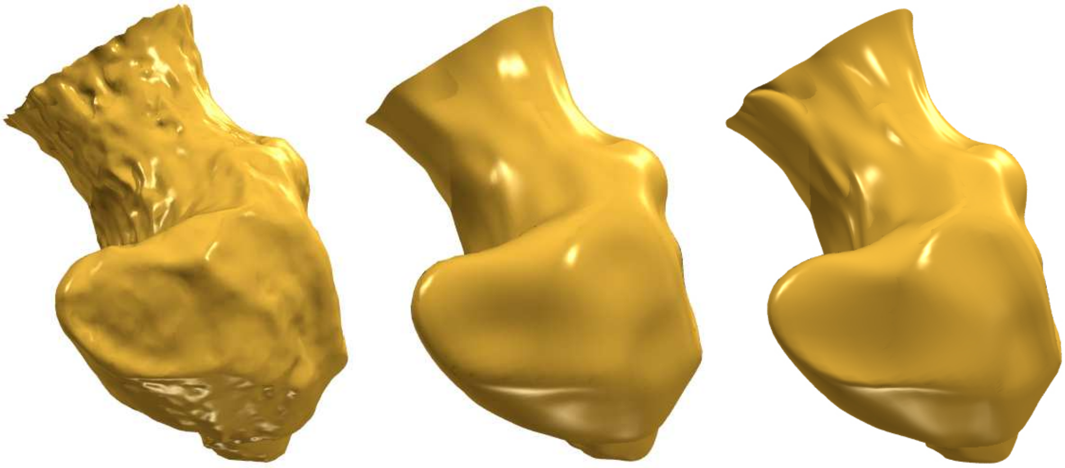

Fig. 8. Fitting of femur part (left) using a periodic B-spline surface without(center) and with (right) feature sensitivity, $w = 0.20$.

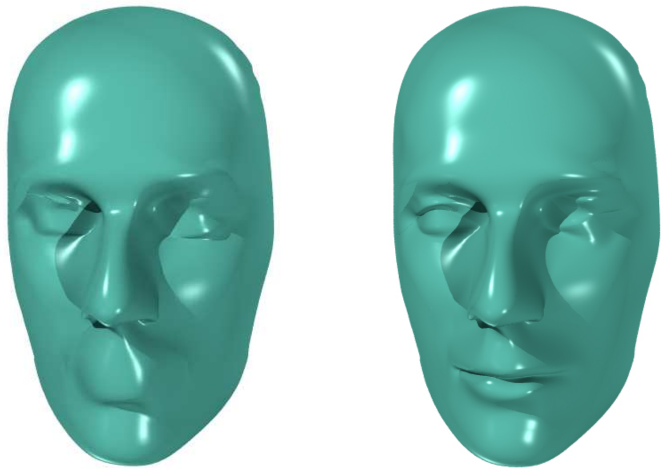

Fig. 9. Fitting results of SDM optimization, using as initial position a fitting surface which has been computed without (left) and with (right) feature sensitivity.

4 Conclusion and Future Research

We have proposed to use a feature sensitive parametrization in connection with a uniform knot distribution for least squares fitting with B-spline surfaces. Even complicated data sets can be fitted well by a single B-spline patch with this method. A single patch is not sufficient for very complex data sets and for objects with a complicated topology. Future work could address this problem by using the fs metric and tools from topology for an automatic patch layout algorithm. Another interesting topic for future research would be variational surface design based on minimization of A_f. A_f favors developable shapes, but also punishes singularities, which are a main problem in developable surface fitting.

Fig. 10. Fitting a car part (left) by a cubic B-spline surface with 20×30 control points and a fs parametrization (center); the result can be further improved with SDM (right).

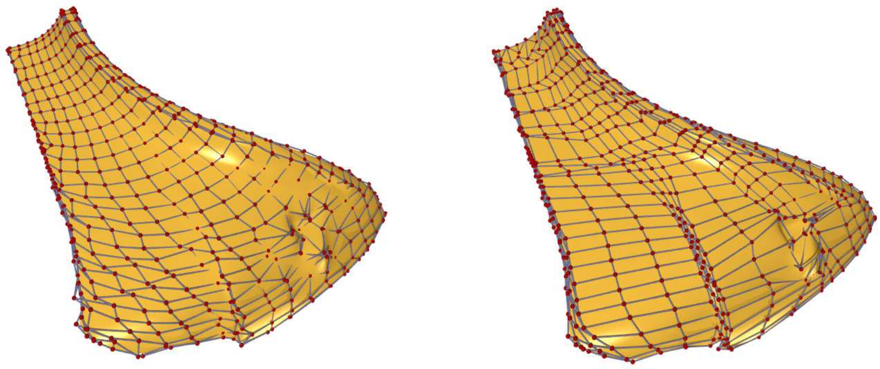

Fig. 11. Control grids obtained by fs fitting before (left) and after (right) SDM optimization for the B-spline surfaces in Figure 10, middle and right.

Acknowledgements

Part of this research has been carried out within the Competence Center *Advanced Computer Vision* and has been funded by the K*plus* program. This work was also supported by the Austrian Science Fund (Grant No. S9206), the Natural Science Foundation of China (Project Numbers 60225016, 60333010, 60321002) and the National Basic Research Project of China (Project Number 2002CB312101). The third author gratefully acknowledges support by Tsinghua University; the friendly and stimulating atmosphere during his stay in Beijing and many fruitful discussions with members of the research group of Shi-Min Hu have greatly promoted this work and other joint research projects.

References

[1] Pierre Alliez, David Cohen-Steiner, Olivier Devillers, Bruno Lévy, and Mathieu Desbrun. Anisotropic polygonal remeshing. In *ACM SIGGRAPH 2003*, pages 485–493, 2003.

[2] Mario Botsch and Leif Kobbelt. Resampling feature and blend regions in polygonal meshes for surface anti-aliasing. *Computer Graphics Forum*, 20(3):402–410, 2001.

[3] U. Clarenz, M. Griebel, M. Rumpf, M. A. Schweitzer, and A. Telea. Feature sensitive multiscale editing on surfaces. *Visual Computer*, 20(5):329–343, 2004.

[4] Ulrich Clarenz, Nathan Litke, and Martin Rumpf. Axioms and variational problems in surface parameterization. *Computer-Aided Geom. Design*, 21(8):727–749, 2004.

[5] David Cohen-Steiner, Pierre Alliez, and Mathieu Desbrun. Variational shape approximation. In *ACM SIGGRAPH 2004*, pages 905–914, 2004.

[6] M. S. Floater. Mean value coordinates. *Computer-Aided Geom. Design*, 20(1), 2003.

[7] M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. In N. A. Dodgson, M. S. Floater, and M. A. Sabin, editors, *Advances in Multiresolution for Geometric Modelling*, pages 157–186. Springer, Berlin, Heidelberg, 2005.

[8] Klaus Hildebrandt and Konrad Polthier. Anisotropic filtering of non-linear surface features. *Computer Graphics Forum*, 23:391–400, 2004.

[9] Josef Hoschek and Dieter Lasser. *Fundamentals of Computer Aided Geometric Design*. A. K. Peters, 1993.

[10] T. R. Jones, F. Durand, and M. Desbrun. Non-iterative, feature preserving mesh-smoothing. In *ACM SIGGRAPH 2003*, pages 943–949, 2003.

[11] Ron Kimmel, Ravi Malladi, and Nir Sochen. Images as embedded maps and minimal surfaces: movies, color, texture and volumetric medical images. *Intl. J. Computer Vision*, 39:111–129, 2000.

[12] Leif P. Kobbelt, Mario Botsch, Ulrich Schwanecke, and Hans-Peter Seidel. Feature sensitive surface extraction from volume data. In *ACM SIGGRAPH 2001*, pages 57–66, 2001.

[13] W. Li, S. Xu, G. Zhao, and L. P. Goh. Adaptive knot placement in b-spline curve approximation. *Comp. Aided Design*, 37:791–797, 2005.

[14] W. Ma and J. Kruth. Parameterization of randomly measured points for least squares fitting of B-spline curves and surfaces. *Comp. Aided Design*, 27:663–675, 1995.

[15] Yutaka Ohtake, Alexander Belyaev, and Hans-Peter Seidel. Ridge-valley lines on meshes via implicit surface fitting. In *ACM SIGGRAPH 2004*, pages 609–612, 2004.

[16] H. Pottmann, S. Leopoldseder, M. Hofer, T. Steiner, and W. Wang. Industrial geometry: recent advances and applications in CAD. *Computer Aided Design*, 37:751–766, 2005.

[17] Helmut Pottmann and S. Leopoldseder. A concept for parametric surface fitting which avoids the parameterization problem. *Computer Aided Geometric Design*, 20:343–362, 2003.

[18] Helmut Pottmann, Tibor Steiner, Michael Hofer, Christoph Haider, and Alan Hanbury. The isophotic metric and its application to feature sensitive morphology on surfaces. In *Proceedings of ECCV 2004, Part IV*, volume 3021 of *Lecture Notes in Computer Science*, pages 560–572. Springer, 2004.

[19] P. Sander, J. Snyder, S. Gortler, and H. Hoppe. Texture mapping progressive maps. In *ACM SIGGRAPH 2001*, pages 409–416, 2001.

[20] P. V. Sander. *Sampling-Efficient Mesh Parametrization*. PhD thesis, Harvard University, May 2003.

[21] P. V. Sander, Steven J. Gortler, John Snyder, and Hugue Hoppe. Signal-specialized parameterization. In *Proc. Eurographics Workshop on Rendering 2002*, pages 87–100, 2002.

[22] G. Tewari, J. Snyder, P. Sander, S. Gortler, and H. Hoppe. Signal-specialized parameterization for piecewise linear reconstruction. In *Eurographics Symposium on Geometry Processing 2004*, pages 57–66, 2004.

[23] Tamas Varady and Ralph Martin. Reverse engineering. In *Handbook of CAGD*, pages 651–681. North Holland, 2002.

[24] J. Vorsatz, C. Roessl, Leif Kobbelt, and Hans-Peter Seidel. Feature sensitive remeshing. *Computer Graphics Forum*, 20(3):393, 2001.

[25] V. Weiss, L. Andor, G. Renner, and T. Varady. Advanced surface fitting techniques. *Computer Aided Geom. Design*, 19:19–42, 2002.

Appendix

Proof of Theorem 1. It is sufficient to employ a principal curvature parametrization $\mathbf{x}(u,v)$ of Φ. Furthermore, let $\mathbf{n}(u,v)$ be a unit normal vector field of Φ. Under these assumptions, one of the coefficients g_{ij} of the first fundamental form vanishes, $g_{12}=0$. Moreover, the coefficients l_{ij} of the so-called third fundamental form (we write partial derivatives via indices, e.g., $\mathbf{n}_u = \partial\mathbf{n}/\partial u$),

$$l_{11} = \mathbf{n}_u^2,\ l_{12} = \mathbf{n}_u \cdot \mathbf{n}_v,\ l_{22} = \mathbf{n}_v^2, \tag{2}$$

are related to the g_{ij}'s via

$$l_{11} = \kappa_1^2 g_{11},\ l_{22} = \kappa_2^2 g_{22},\ l_{12} = g_{12} = 0. \tag{3}$$

The area element of Φ is given by

$$dA = \sqrt{g_{11}g_{22} - g_{12}^2}\, dudv = \sqrt{g_{11}g_{22}}\, dudv. \tag{4}$$

Likewise, the area of the image manifold Φ_f, whose parametrization is $\mathsf{X}(u, v) = (\mathbf{x}(u, v), w\mathbf{n}(u, v))$ is found via

$$\begin{aligned} A_f &= \int \sqrt{\mathsf{X}_u^2 \mathsf{X}_v^2 - (\mathsf{X}_u \cdot \mathsf{X}_v)^2}\, dudv \\ &= \int \sqrt{(g_{11} + w^2 l_{11})(g_{22} + w^2 l_{22}) - (g_{12} + w^2 l_{12})^2}\, dudv. \end{aligned}$$

Using (3) and (4), this simplifies to the form stated in (1),

$$A_f = \int \sqrt{(1 + w^2\kappa_1^2)(1 + w^2\kappa_2^2)g_{11}g_{22}}\, dudv = \int \sqrt{1 + w^2(\kappa_1^2 + \kappa_2^2) + w^4 K^2}\, dA.$$