Vision, Video and Graphics (2005)
E. Trucco, M. Chantler (Editors)

Merging Graphics and Vision for 3D Face Recognition

Li Bai and Yi Song
School of Computer Science & IT
University of Nottingham

Abstract

This paper presents a new approach to automatic 3D face modelling from unstructured point cloud data. An efficient B-Spline surface-fitting algorithm is used to obtain an initial parametric surface for each face point cloud data set. Knot vectors for each individual face surface are then standardised to produce a set of uniform knot vectors so that all the surfaces can be seen as fitted with the same set of knot vectors. Mapping from object space to shape space can then be established so that each 3D face can be described by a small number of shape descriptors. The use of shape descriptors allows automatic registration between face models. More importantly, it allows dynamic facial variation to be modelled and analysed via 3D warping, resulting in a powerful approach to quantifying the differences among individuals required for face recognition. 3D warping is often used in simulations in computer graphics. This paper explains, for the first time, how 3D warping can be exploited for face recognition based on multi-resolution analysis of warping fields. The methodology allows the quantitative study of variation in characteristics previously only described from a qualitative perspective.

Categories and Subject Descriptions: I.4.5 [Image Processing and Computer Vision]: Reconstruction

1. Introduction

Though facial recognition has been a subject of intense study, considerable challenges remain to develop a system that is accurate and robust enough for real applications. Problems identified are largely due to variations in lighting conditions and head poses. To overcome the problems many researchers have been researching 3D facial verification algorithms, as a 3D model contains all possible views of a face and may handle pose and lighting variation better. Theoretical and technical advances in 3D data capture and modeling techniques, and the increasing affordability of the products also present the opportunity for 3D facial recognition at security critical sites. However, a breakthrough solution has so far eluded the best companies and researchers in the world.

Mathematical transform plays an important role 3D facial recognition. This usually relies on establishing some landmark correspondence between two faces. For example, Procrustes analysis is a way of comparing shapes by calculating a similarity transform between shape landmarks so that they are as close to each other as possible according to the Euclidean distance. However, it is not always possible to find an exact mathematical transform between two shapes, as the number of landmarks often exceeds the dimensions of the coordinate system. Correspondence between landmarks also requires an iterative optimisation procedure.

Statistical models assume a training set of images in which landmark points have been manually marked. With enough training examples such models may be used to synthesize any image of normal anatomy and locate all the structures represented by the model in the target image. Tim Cootes pioneered the use of active contour models to model shape deformation [CT94] [CET98]. This is based on a point distribution model (PDM) estimated from a training set. The mean point positions and their modes of variation are used to restrict the deformation process. The method requires a large training set, and manual marking corresponding landmarks on every image in the training set. Though in theory this might be extended to 3D, in practice obtaining good (dense) correspondences between 3D models is still an active research topic [CT01]. Kelemen et al. [KSG99] [SKB*96] closely follow the work of Taylor and Cootes but explore an alternative approach to obtain statistical models of shape and appearance. Their method is based on a hierarchical parametric object description rather than a point distribution model. However, this only works on closed surfaces with spherical topology, it cannot be generalised to 3D open surfaces easily, which is the common case for face modelling.

In summary, current approaches to facial recognition match two shapes based on a limited number of landmark correspondence between shapes. This is far from sufficient for describing the subtle differences between complex objects such as faces. None of the approaches can represent shape differences explicitly or in sufficient local detail (e.g., every point). The fact is that recognising a face is to distinguish it from any other. This can only be done efficiently if we explicitly quantify the difference between faces. In other words, we need to deal with facial variability. We need a method that can automatically quantify variations for any point (rather than just a few pre-defined landmarks) on face models. We also need to be able to analyse hidden trends/patterns in the resulting continuous difference maps or deformation fields. In this paper, we describe a new approach that will

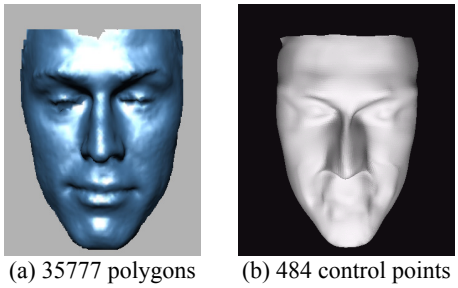
address these issues. Main contributions of this work are:

- Automatic construction of uniform parametric face models from points clouds using B-Splines. Knot vectors used to fit individual faces are standardised to allow uniform representation of all faces in a parametric space, and detailed analysis of their physical properties.
- Automatic and non-iterative registration of 3D faces. Instead of matching points on two surfaces directly, we register objects in the parametric space using shape descriptors. This advances the state of art using iterative methods to register objects by minimising the distance between corresponding landmarks.
- Multi-resolution wavelet analysis of warping fields. The aim is to quantify shape difference at every location and different scales by warping one object to another and analysing the warping field using wavelets. This allows a more effective portrayal of differences among individuals.

The outline of this paper is as follows: in Section 2, we detail the procedure of automatic construction of 3D models using B-Splines. In Section 3, non-iterative registration is proposed. Section 4 explores face recognition based on facial variation using 3D face warping. Finally, we conclude the paper in Section 5.

2. Automatic Construction of 3D Face Models

3D objects are usually described by polygon meshes. However, polygonal representations are more suitable for object rendering rather than concisely describing 3D objects for recognition. We represent face models using shape descriptors (a small number of parameters) derived from B-Spline surface interpolation of point clouds. A comparison is made by rendering the same 3D face but with (a) by polygons and (b) by shape descriptors in Figure 1. We use shape descriptors of size 484 to represent the anatomical information of a face, whilst 35777 polygons are needed to produce a similar result.



(a) 35777 polygons (b) 484 control points

Figure 1 *Polygonal and parametric representations*

B-Spline modelling generally falls into two categories. One approach is to reconstruct a B-Spline surface from meshes [EH95] [KL96] [Loo95]. Another is to model objects by *hierarchical B-Spline refinement* which interpolates a data grid using a *control lattice hierarchy* [HDD92] [LWS97]. Eck and Hoppe introduced an automatic reconstruction of B-Spline surfaces from meshes, which require no user intervention to label boundary points or draw boundary curves on a surface. While this method produces high quality surfaces, it requires a number of expensive optimisation steps and heavy computation. An example of data interpolation using a control lattice hierarchy is Lee's multilevel B-Spline model. In contrast with previous approaches by Forsey and Bartels [FB95] [FB98], Lee's approach can handle scattered data. However, multilevel B-Spline approximation cannot reconstruct a non-planar surface exactly. On the other hand, the algorithm is not affine invariant and the shape approximation accuracy depends on the density of control lattice. Since data points are projected to underlying control lattices of varying resolutions, the relative positioning between data points and the lattices affects the approximation function. Our approach overcomes these limitations. We allow 3D surface to be constructed directly from the unstructured point cloud. No control lattices are required.

2.1 B-Spline Surface Fitting

We use B-Spline surface fitting of unstructured point clouds. Since B-Spline surfaces are tensor product surfaces, a surface-fitting problem can be reduced to a sequence of curve fitting processes, making computation particular efficient. However, almost all the previous approaches were restricted on grid data interpolation. For example, the data used by Schmitt et al. [SB96] was organised in a rectangular array. In order to adapt this algorithm to unstructured data, a new approach to standardising the set of parameters associated with each curve to a uniform parameter set is proposed in session 2.2. Therefore, the original data can be decomposed into small portions upon which the curve-fitting process is applied independently. After standardisation, all the surface-fitting processes use the same uniform parameter set. More detailed discussions are given below.

A B-Spline curve of degree p is defined in the parametric domain $u \in [0, 1]$ by weighted control points $Pi \in R3$

$$C(s) = \sum_{i=0}^{m} N_{i,p}(s) P_i \qquad (1)$$

A B-Spline basis function Ni,p(u) defined over knot sequence $U=\{u_0, u_1, \ldots, u_{m+p+1}\}$ can be expressed as:

$$N_{i,0} = \begin{cases} 1 & \text{if } u_i \le s < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(s) = \frac{s-u_i}{u_{i+p}-u_i} N_{i,p-1}(u) + \frac{u_{i+p+1}-s}{u_{i+p+1}-u_{i+1}} N_{i+1,p-1}(s) \qquad (2)$$

Each curve is interpolated separately over its own knot vector. We apply different automatic sampling schemes on different parts of a face. For example, the forehead area is rather flat with little curvature changes. A sparse and evenly sampling scheme will work well. On the

contrary, the area close to the nose contains sharp curvature variations. Thus a dense and uneven sampling scheme is necessary to guarantee precise interpolation. The result from the curve set interpolation leads to surface fitting.

$$S(s,t)=\sum_{i=0}^{m}\sum_{j=0}^{n}N_{i,p}(s)N_{j,q}(t)P_{i,j} \quad (3)$$

$$P_{i,j}=\begin{pmatrix}P_{i,j}^{x}\\P_{i,j}^{y}\\P_{i,j}^{z}\end{pmatrix}$$

A B-Spline surface of degree (p,q) is defined in the parametric domain (s, t) ∈ [0, 1]. Here $P_{i,j} \in R3$ denote the control points of the surface, Ni,p(s) are the B-Spline basis functions of degree p in the u-direction, defined over the knot vector $U=\{u_0, u_1, \ldots, u_{m+p+1}\}$, and Nj,q(t) are defined analogously over the knot vector $V=\{v_0, v_1, \ldots, v_{n+q+1}\}$ in the v-direction. From equations (2) and (3), it can be observed that each 3D face is modelled upon its own pair of knot vectors Ui and Vi (the subscript indicates ith individual in the data set).

2.2 Parametric Representation

After surface fitting, we obtain a set of control points $\{P_{i,j}\}$ for each model. From equation (3), it is clear that any point on the surface can be calculated from the control points and the basis functions. Let x, y and z denote Cartesian coordinates in object space and s and t constitute the parametric space. f ∈ R3 is a surface point corresponding to a pair of parameter coordinates (s, t). Control points are 3D vectors with components (x_k, y_k, z_k). As the basis functions are defined by the individual specified knot vectors over the parameter space, each individual will have a different set of basis functions B(Ui(s), Vi(t)) where Ui and Vi specify the knot vectors for the ith individual. Thus the relationship between parametric space and object space is

$$f(s,t)=\begin{pmatrix}x(s,t)\\y(s,t)\\z(s,t)\end{pmatrix}=\sum_{k=0}^{K}B_k(U_i(s),V_i(t))\begin{pmatrix}x_k\\y_k\\z_k\end{pmatrix} \quad (4)$$

In order to obtain a direct mapping from object space to parametric space, the dependency on individual knot vectors must be removed. We propose a technique for standardising individual knot vectors to produce a set uniform knot vectors so that all the face models can be surface fitted against the same set of knot vectors.

Let Ug denotes the uniform knot vector in u-direction; Ui and Uj are the individual specified knot vectors in u-direction (ith and jth individual in the data set which has $\{0\} \le (Ui \cap Uj) < (Ui \cup Uj)$. The simplest way to make Ui = Uj is to apply the knot insertion algorithm [WW92] to unify Ui and Uj (i.e. Ug = Ui ∪ Uj). Consequently the size of Ug will be too big to be acceptable in real applications (i.e. $U_g=\bigcup_{i=0}^{n}U_i$ where n is the number of individuals in the data set). Instead of simply merging all knot vectors together, we pre-define a Ug of size G. For each knot vector Ui of size K ≠ G. We traverse each element in Ui and Ug to keep Uia ($0 \le a \le K$) untouched if it is also an element in Ug. If ∃b. (Ugb ∈ Ug) ^ (Ugb ∉ Ug) ($0 \le b \le G$), we insert Ugb into Ui during the re-parameterisation process.

After the standardisation process, we will have Ui = Ug of size G without changing the shape of the surface (curves). This process is repeated in the v-direction. Independent basis functions of the knot vectors can be obtained. The new parameters set obtained from the standardisation process defines the shape descriptors $\{Q_g\}=(\hat{x}_g, \hat{y}_g, \hat{z}_g)$. After this an one to one mapping function can be represented as

$$f(s,t)=\begin{pmatrix}x(s,t)\\y(s,t)\\z(s,t)\end{pmatrix}=\sum_{g=0}^{G}\hat{B}_g(s,t)*\begin{pmatrix}\hat{x}_g\\\hat{y}_g\\\hat{z}_g\end{pmatrix} \quad (5)$$

It is clear from equation (5) that when the parameters s and t run over their domain (i.e. s = 0 … 1, t = 0 … 1), f(s,t) runs over the whole 3D surface, i.e. one to one mapping exists between Cartesian coordinates of object space and parametric coordinates (s, t) of the shape space. The shape descriptors are shown in Figure 2, while the corresponding surfaces reconstructed from the shape descriptors are showed in Figure 3.

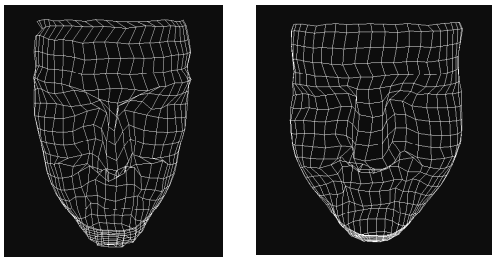


Figure 2 *Meshes of 484 shape descriptors*

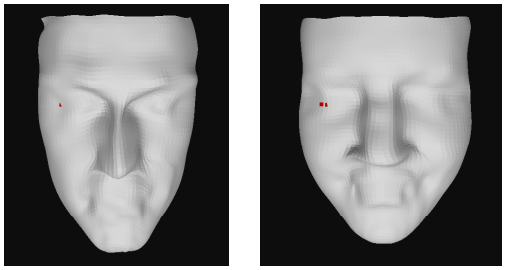


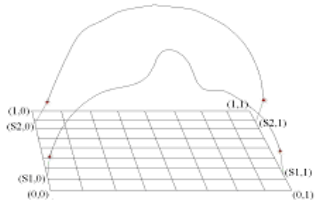
Figure 3 *Surface reconstructed from shape descriptors*

3 Registration

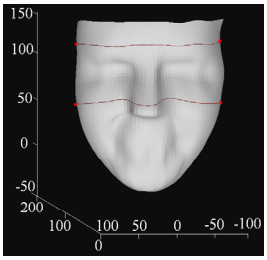
The shape representation results in a continuous mapping between similar object surfaces. As any point on such a surface can be mapped to exactly one point in the parameter space and vice versa, by uniformly

sampling parameter space, corresponding pairs of surface points can be generated, i.e., two points on to surfaces are matched if they have the same parameter value. However, there is no guarantee here that the corresponding points are located in the same area across objects, though surface boundary points will be mapped to the boundary of the parameter space. An example is given in Figure 4, where two curves on the surface are plotted, with starting points of the curves mapped to (s1,0), (s2, 0); and end points mapped to (s1,1), (s2,1), s1, s2 ∈ [0,1] respectively.

We have developed an automatic correspondence method, which guarantees consistent mapping between models. Basic facial features used for this are the mid-plane of the face. Since the nose tip could be located at the stage when point clouds data is generated, we make use of this information. The detection of the mid-plane is a two-step searching scheme. First, a symmetric analysis is applied to estimate the rough position of the saddle point of nose (point between eyes) and the rotation of the point cloud about the in y-axis. Second, the possible areas containing the inner corner of eyes can be located. Combining with the texture information, the inner corner of eyes can be located precisely. Then we can calculate the precise location of the saddle point and the head pose. We can then transform the data to a canonical space where rotations with respect to the x, y, and z axes are zero. Figure 5 illustrates the face aligned in the canonical space. The corresponding points should now be located in similar regions across objects, see Figure 6.

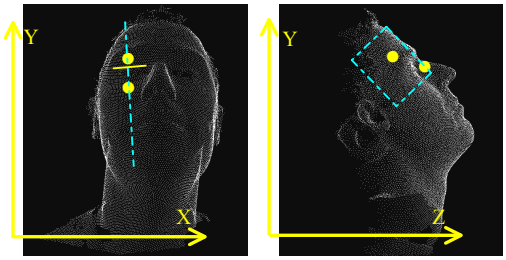


(a) Parameter space

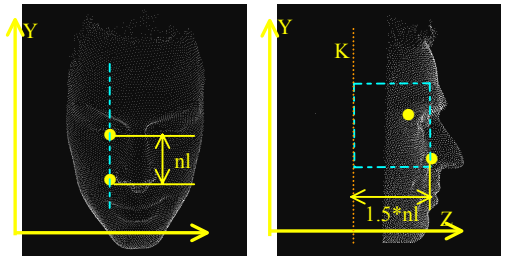


(b) Object space

Figure 4 *Relationship between object space and parameter space*



(a) Points clouds



(b) Normalised point clouds

Figure 5 *Nomalising point cloud data*

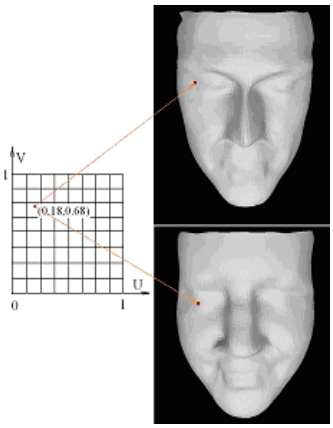
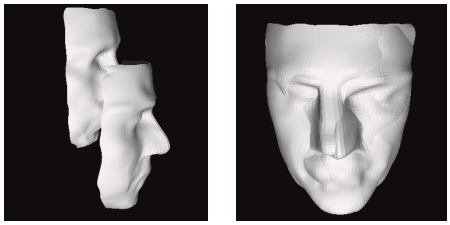


Figure 6 *Correspondence in the parametric space*

Once correspondences between models are established in the canonical space, registration between two models can take place. Most existing approaches use an iterative procedure, which minimises the distance between corresponding points on two objects. In contrast, we register objects using shape descriptors, which have affine invariant property, i.e. the parameterised surface will not change its geometry when geometric transformations such as translation, rotation and scale are taken on the shape descriptors. Therefore we can manipulate the shape descriptors directly to align two models. Figure 7 shows two face models before (a) and after (b) normalisation, two face models are overlapped.



(a) Before (b) After

Figure 7 *Registration*

4. Application to Face Recognition

In this section we introduce a method based on 3D warping for 3D face recognition. We first create a database of 3D face models. This uses a 3D scanner to generate point clouds and the method described in previous sections to represent each face by a set of shape descriptors. These shape descriptors are especially useful for automatic 3D warping between objects, see Figure 8.

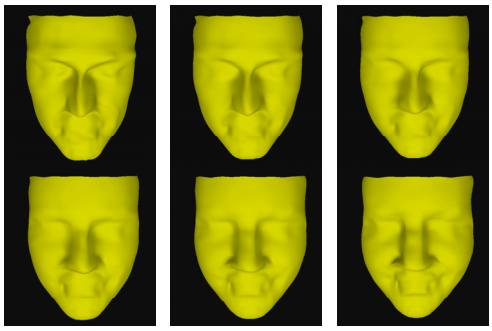
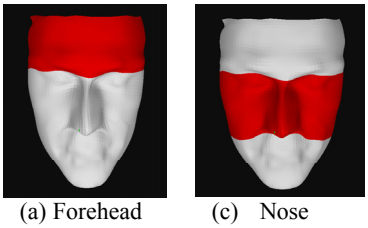


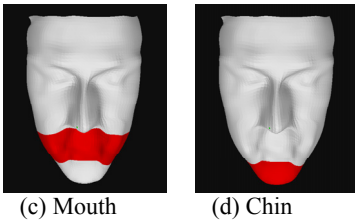
Figure 8 *Face warping*

Warping one face to another results in a warping vector field, which can be used to measure the similarity of objects, i.e. if two faces are similar, the 'cost' of warping/transforming one face to another would be less, and vice versa. This suggests a new 3D face recognition method: Taking one model as a generic model, we can calculate how each of the other face models deviates from the generic model. At the recognition stage we can work out the deviation of an input face model from the generic model, and use this information as an index into the face database. We can also analyse facial differences by analysing the warping field between two faces.

For efficiency and accuracy, we use a hierarchical scheme, dividing a face to four parts, see Figure 9: forehead (9a), nose (9b), mouth (9c) and chin area (9d). With the help of the shape descriptors different areas can be unambiguously marked in the parametric space.



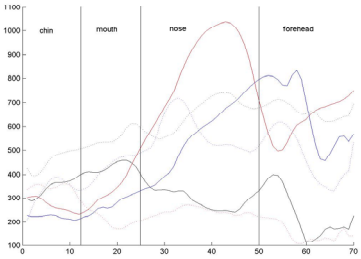
(a) Forehead (c) Nose



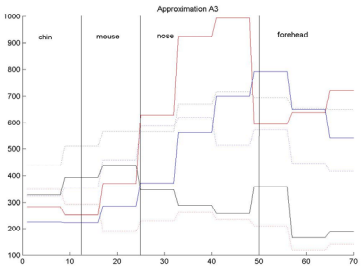
(c) Mouth (d) Chin

Figure 9 *Facial components*

Figure 10 (a) plots the deformation vectors generated by warping 6 face models to the generic face in turn. Wavelet analysis is employed to provide further reduction in dimensionality of the deformation vectors. Wavelet coefficients extracted from the deformation vectors are illustrated in Figure 10 (b). These coefficients can be used for classification of faces.

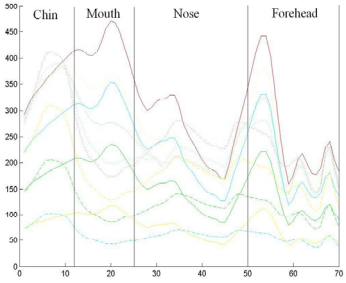


(a) Deformation vectors

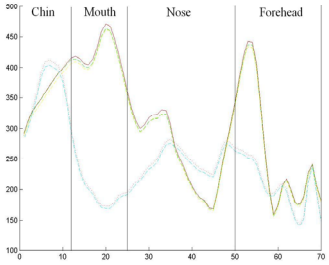


(b) 3rd level approximation

Figure 10 *Wavelet analysis*



a) Difference analysis



(b) Matching faces

Figure 11 *Comparing similar faces*

The ultimate challenge for face recognition is to be able to distinguish two similar looking people. We have no such data available to use so we use linear warping to obtain similar looking faces. For example, for a pair of models M_1 and M_2, we have 3 intermediate faces created by warping from M_1 to M_2 with 25%, 50% and 75% variance respectively. Figure 11 (a) displays the vector field analysis of 12 models created. X-axis represents the locations of the vector fields in the parametric space. Y-axis represents the differences between the models and the generic model. In Figure 11 (b), we illustrate matching result. We match two new faces against existing models to see the differences.

5. Conclusion

This research represents a promising step in designing a system that is capable of modelling facial variations for accurate face recognition. Shape descriptors are able to represent 3D objects in a very compressed format. More importantly, these allow dynamic facial variation to be modelled, resulting a powerful approach to 3D facial database search and to quantifying individual differences for identification. The processes from the surface fitting to obtaining the parametric descriptions are fully automatic. The system is tested on a personal computer of Pentium 4/512M RAM. Compared with an existing closed surface parameterisation method, which requires 33s to 536s processing time depending on the complexity of the objects [BGK96], our method takes only one second.

6. References

[BGK96] C. Brechbuhler, G. Greig, O. Kubler, Parametrization of Closed Surfaces for 3-D Shape Description, March 29, 1996.

[CET98] T. Cootes, G.Edwards, C. Taylor, Active Appearance Models, in Proc. European Conference on Computer Vision 1998, Vol. 2, pp. 484 – 498, Springer, 1998.

[CT94] T. Cootes, C. Taylor, Modelling Object Appearance Using the Grey-level Surface, Proc. British Machine Vision Conference, pp. 479 – 488, 1994.

[CT01] T. F. Cootes, C. J. Taylor, Statistical Models of Appearance for Medical Image Analysis and Computer Vision, in Proc. SPIE Medical Imaging, 2001.

[EH95] M. Eck, H. Hoppe, Automatic Reconstruction of B-Spline Surfaces of Arbitrary Topological Type, 1995.

[KL96] V. Krishnamurthy, M. Levoy, Fitting Smooth Surfaces to Dense Polygon Meshes, ACM-0-89791-746-4/96/008, 1996.

[FB95] D. Forsey, R. Bartels, Surface Fitting with Hierarchical Splines, ACM Transactions on Graphics, Vol. 14, No. 2, April 1995, pages 134-161.

[FB98] D. Forsey, R. Bartels, Hierarchical B-Spline Refinement, Computer Graphics, Vol. 22, No. 4, August 1998.

[HDD92] H. Hopper, T. DeRose, T. Duchamp, Surface reconstruction from unorganised points, Computer Graphics (SIGGRAPH'92) 26, 2, 1992.

[KSG99] A. Kelemen, G. Szekely, G. Gerig, Elastic Model-based Segmentation of 3D Neuroradiological Data Sets, IEEE Transactions on Medical Imaging, Vol. 18, No. 10, October 1999.

[Loo95] C. Loop, Apple Computer, Smooth Spline Surfaces over Irregular Meshes, Inc. 1995.

[LWS97] S. Lee, G. Wolberg, S. Y. Shin, Scattered Data Interpolation with Multilevel B-Splines, IEEE Transactions on Visualization and Computer Graphics, Vol. 3. No. 3. July-September 1997.

[SB96] F. Schmitt, B. Barsky, An Adaptive Subdivision Method for Surface-fitting from sampled data. In processing of SIGGRAPH86. Comput. Graph. 20, 4, pp 179-188.

[SKB*96] G. Szekely, A. Kelemen, C. Brechbuhler, G. Gerig, Segmentation of 2D and 3D objects from MRI volume data using constrained elastic deformations of flexible Fourier contour and surface models, Medical Image Analysis, Vol. 1, No. 1, pp 19-34, 1996.

[WW92] A. Watt, M. Watt, Advanced Animation and Rendering Techniques, ACM press, 1992.