Application of Radial Basis Functions for CAD and CG

Vladimir Savchenko
Hosei University, Japan
vsavchen@k.hosei.ac.jp

Abstract

Despite their long history, radial basis functions have never really become a widely used tool for surface generation and image/surface modifications. This paper presents work in progress, and continues a project devoted to developing a system for shape modeling based on implementation of RBF technology. Experimental results are included to demonstrate the functionality of our mesh-modeling tool. In particular, we consider such applications as surface reconstruction, surface retouching, animation, and shape smoothing. Also we discuss an algorithm for local mesh generation and polygon simplification.

Keywords: *radial basis functions, reconstruction, surface modification.*

1. INTRODUCTION

A challenging goal in computer graphics (CG) and computer aided design (CAD) is to provide powerful technique for modeling the shape of an object. Indeed, when we are using only generic properties such as position of a point of the object that is deformed the problem of constructing smooth surface satisfying certain constraints can be formulated as a mapping function from $\mathbf{R}^3$ to $\mathbf{R}^3$. Such a space-mapping technique based on RBFs is a powerful tool, which offers simple and quite general control of simulated shapes. In fact, a model of extended space mapping [1] is used and incorporates geometric space mappings and function mappings from $\mathbf{R}^3$ to $\mathbf{R}$. Constructive solid geometry (CSG) is usually used in many CAD applications. Traditionally, CSG modeling uses simple geometric objects for a base model, which can be further manipulated by implementing a certain collection of operations such as set-theoretic operations, blending, or offsetting. The operations mentioned above and many others have found quite general descriptions or solutions for geometric solids represented as points (x,y,z) in space satisfying $f(x,y,z) \geq 0$ for a continuous function f. Radial basis functions (RBFs) offer a mechanism to obtain extrapolated points of a surface for various parts of a reconstructed object that can be used as “CSG components” to design a volume model. Nevertheless, we mainly consider RBFs as a tool for surface modifications based on a mapping from $\mathbf{R}^3$ to $\mathbf{R}^3$.

2. OVERVIEW

In spite of a flurry of activity in the field of scattered data reconstruction and interpolation, this matter remains a difficult and computationally expensive problem. A vast amount of literature is devoted to the subject of scattered data interpolation methods and their applications, see, for instance [2][3][4][5][6][7]. However, the required computational work is proportional to the number of grid nodes and the number of scattered data points. Special methods to reduce processing time were developed for thin plate splines and discussed in [8][9], see also recent publications [10][11][12].

Here, we shall give a short account of the shape transformation method used in the applications considered in this paper. To interpolate the overall displacement, we use a volume spline based on Green's function, for more references, see [13]. This is well known problem - to find an interpolation spline function $u \in W_2^m(\Omega)$, where $W_2^m(\Omega)$ is the space of functions whose derivatives of order $\leq m$ are square-integrable over $\Omega \subset \mathrm{R}^n$, such that the following two conditions are satisfied: (1) $u(p_i) = h_i$, $i = 1,2, \ldots ,N$, and (2) u minimizes the bending energy, if the space transformation is seen as an elastic deformation. For an arbitrary three-dimensional area Ω, the solution of the problem is well - known: the volume spline $f(P)$ having values h_i at N points P_i is the function

$$f(P) = \sum_{j=1}^{N} \gamma_j \phi(|P - P_j|) + p(P), \qquad (1)$$

where $p = v_0 + v_1 x + v_2 y + v_3 z$ is a degree one polynomial. To solve for the weights γ_j we have to satisfy the constraints h_i by substituting the right part of equation (1), which gives

$$h_i = \sum_{j=1}^{N} \gamma_j \phi(|P_i - P_j|) + p(P_i). \qquad (2)$$

Solving for the weights γ_j and v_0, v_1, v_2, v_3 it follows that in the most common case there is a doubly bordered matrix $\boldsymbol{T}$, which consists of three blocks, square sub-matrices $\boldsymbol{A}$ and $\boldsymbol{D}$ of size $N \times N$ and 4×4 respectively, and $\boldsymbol{B}$, which is not necessarily square and has the size $N \times 4$.

Since the RBF $\phi(r)$ is not compactly supported, the corresponding system of linear algebraic equations is not sparse or bounded. Storing the lower triangle matrix requires $O(N^2)$ real numbers, and the computational complexity of a matrix factorization is $O(N^3)$. Thus, the amount of computation becomes significant, even for a moderate number of points. Wendland in [14] constructed a new class of positive definite and compactly supported radial functions for 1D, 3D and 5D spaces of the form $\phi(r) = \psi(r)$, $0 \leq r \leq 1$; $0, r > 1$, where $\psi(r)$ is a univariate polynomial whose radius of support is equal to 1. Thanks to the efficient octree algorithm proposed in [15], the resulting matrix is a band-diagonal matrix that permits handling of large data sets in a reasonable time.

A space-mapping in $\mathbf{R}^n$ defines a relationship between each pair of points in the original and deformed objects. Nevertheless, heights h_i, are not necessarily arbitrary points of Euclidean space E^n. Our approach presents an attractive possibility of using function mapping for controlling local deformations by placing arbitrary control points inside or outside an initial implicitly defined object G, and they are assumed to belong to the surface of a modified object G^m. Thus, the control points define the deformation of G resulting in G^m.

3. APPLICATIONS

3.1 Surface reconstruction

A vast volume of literature is devoted to the subject of scattered reconstruction and interpolation, see, for instance, an overview in [15]. Methods exploiting RBFs can be devided into three groups. The first group is "naïve" methods, which are restricted to small problems, but they work quite well in applications, dealing with shape transformation. The second group is fast methods for fitting and evaluating RBFs [11]. The third is compactly supported RBFs. Let us notice here about recent outstanding work of Ohtake et al. [16] where compactly supported radial basis functions (CSRBS) are used as blending functions. The significance of this work is that the method proposed enables obtaining high quality reconstruction results and handle realistic amounts of data.

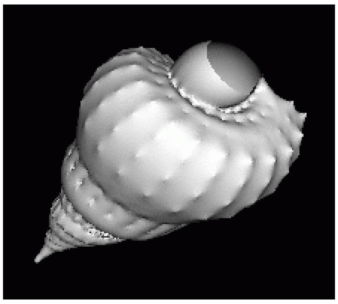


Figure 1: "Seashell" CSRBFs surface reconstruction where sphere as a "carrier" function is used. Number of points: 917. Processing time: 0.56 sec. (includes surface extraction time: 0.41 sec).

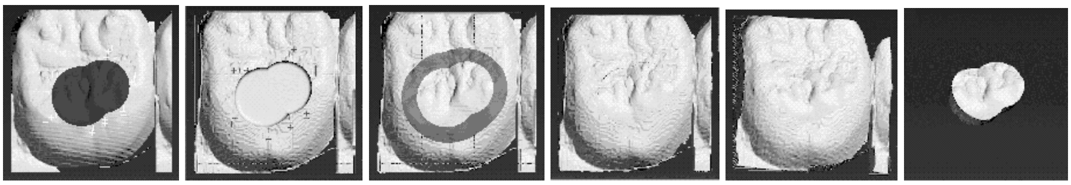
The application shown in Fig. 1 (an example of reconstruction of implicit surfaces (CSG object)) demonstrates applicability of an extremely simple approach proposed in [7] that allows us to attain rather acceptable results. In our software implementation, we employ a standard approach for creating a binary tree from an initial point data set with an additional required parametric value K, which denotes the maximum number of points in a leaf. Such a tree allows to provide an efficient sorting of scattered data [15] that leads to obtaining a band diagonal sub-matrix $\boldsymbol{A}$; after that Cholesky decomposing and block Gaussian solution are applied, as it was proposed by George and Liu in [17].

3.2 Hole filling

Point sets obtained from computer vision techniques are often non-uniform and even contain large missing areas of points. Another source of such a data, for instance, is partly destroyed natural objects, for instance, such as teeth that need a treatment.

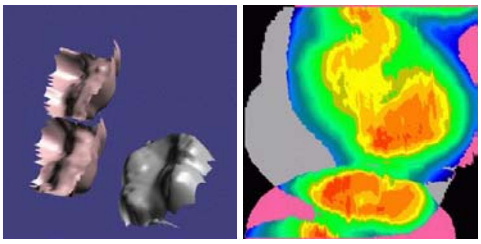
Three approaches to reconstruct missing parts have been dominant in the CAD area: the first one works with 3D polygonal models to stitch damaged or incorrectly calculated nodes of 3D geometric objects, the second one is an approach dealing with fitting of the data generated according to some geometrical features such as curvature, and the third one is actually based on a well-founded mathematically set-level approach. Partial differential equations are widely used to model a surface subject to certain constraints.

Another way is to apply space mapping technique based on the use of RBFs, for more references, see [18]. In the application shown in Fig. 2, the optimization task is to find a functionally transformed occlusal surface (an inlay part) of a model tooth that best matches the remaining occlusal surface of a treated tooth. Actually, this approach can be called "cloning". We suppose that RBFs are suitable for sufficiently moderate 3D data sets. Nevertheless, they possess many features that make them very attractive for CAD applications dealing with modification of geometric objects, see [19], where occlusal surface modeling for restorations, based on jaw articulation simulation was used as shown in Fig.3. Fig. 3(a) shows the complexity of the surface of teeth, which are the subject of correction. Requirements of avoiding of interpenetrations with the opponent teeth and preserving main topological features of the occlusal surface of teeth are imposed on design of the occlusal surface of restoration. Arbitrarily placed points (Fig. 3(b)) induced by the distance distribution are used to control 3D occlusal surface deformations by RBFs.



(a) (b) (c) (d) (e) (f)

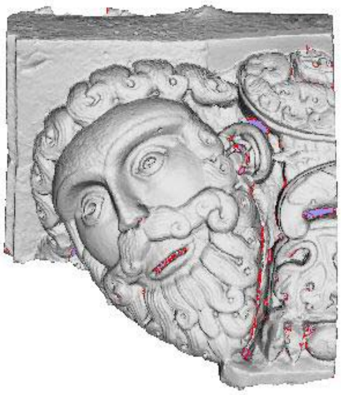
Figure 2: (a) Model tooth. (b) Tooth to be treated. (c) Approximation area (extended, pre-set boundary. (d) Treated tooth after application of genetic optimization. (e) Treated tooth after final refinement based on RBFs. (f) Resulting approximated CSG object.



(a) (b)

Figure 3: (a) Surface of teeth. (b) Distance maps of collision removal by applying RBF mapping function.

In [20] an approach to hole filling of polygonal data was proposed. The algorithm includes holes extraction step, polygon stitching, and a hole surface improvement based on space-mapping technique. For sufficiently small holes, polygon stitching based on the use of dynamical programming demonstrates quite good results as it can be seen in Fig. 4.



(a)

(b)

Figure 4: Example of surface retouching of a real polygonal model**.** (a) "Stoned" model (courtesy of R. Scopigno and M. Calliery of Institute CNUCE). Model size – 88478 points. Red lines show hole areas. (b) Model after surface retouching.

To illustrate the applicability of the space-mapping technique (third step of our algorithm) to the surface-retouching problem, we first show an example of image inpainting in Fig. 5.

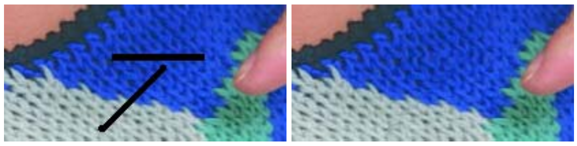


Figure 5: Example illustrating image inpainting approach. "Wool", one additional sloping scratch was added to the test image from [21]. Processing time: 0.1 sec.

The principal contribution of the approach is a surface – retouching algorithm based on a local approximation of missing data. Example in Fig. 6 demonstrates the applicability of the approach for a rather complicated geometric object with 16 holes and the result of completely automatic reconstruction of the missing parts of the object. Let us note that regions where one-to-one mapping or neighborhoods that are not homeomorphic to a disc can be observed in this example.

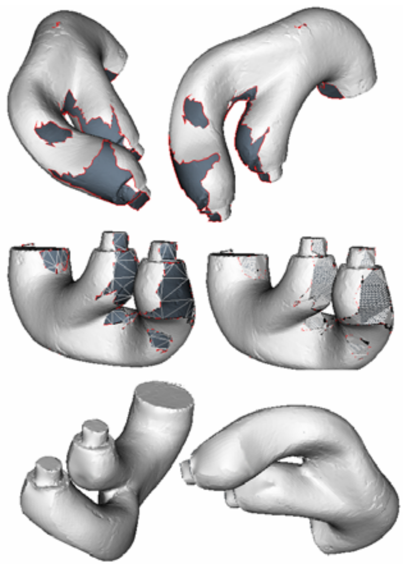
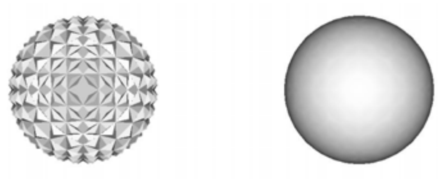


Figure 6: Solution based on CRRBFs. (1) Original model "Port6" (19467 polygons) contains 16 holes. (2) Prediction of the surface inside the holes area – triangulation (left) and linear subdivision of the triangulated surface (right). (3) Result of the extrapolation for the holes area. Total processing time 7.1 sec.

3.3 Surface smoothing

The approach discussed above shows the obvious relationship between the surface-retouching problems and shape smoothing. Fig. 7 presents an example of compactly supported RBF smoothing of polygonal surfaces. The shape-smoothing algorithm exhibits, in practice, good features and volume-preserving properties.



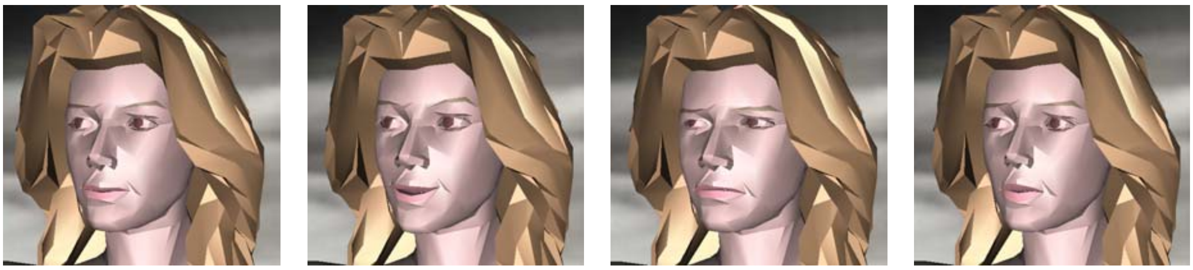
(a) (b)

Figure 7: (a) The original noisy sphere "Epcot" model, (770 vertices, 1536 polygons), (b) smoothed model after 5 iterations based on 11-point interpolation. Processing time: 0.6 sec.

3.4 Real time facial animations

3D geometric modeling systems based on shape deformations have been pursued by many researchers and take mainly advantage of the simple idea that tangible geometry of deformations can be defined by the user assigned starting and destination points. Probably this approach firstly was implemented in the papers of Wolberg [22] and Beier and Neely [23] for 2D morphing.

For real-time applications computing of the transformations is the most critical part in the sense of time optimizations. Fig. 8 shows an example of facial animations by compactly supported RBFs. For every point, which is inside the radius of support distance is calculated one time and after that space transformations are calculated in accordance to a phase parameter (value varies from 0 to 1) that defines total deformation [24]. The deformation process is regarded as taking place step by step so that the transition from a known state to a new state takes place with small increments in deformations. That is, intermediate transformations for every step of animation are generated according to the phase parameter.



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

Figure 8: An example of facial animations (7024 polygons, deformation defined by 32 vectors, radius of support is equal to 0.2): (a) – original model, (b) – "smile", (c) – "upset", (d) – "kiss". Processing time: 107 fps.

3.5 Mesh generation

Surface reconstruction methods can be broadly classified into global and local approaches. From our point of view, methods based on the idea of local reconstruction are promising in CAD and CG applications dealing with huge amounts of scattered data.

The partition of unity method (PUM) for the construction of interpolation and approximation was pioneered by Shepard [25] and was later extended by Franke and Nielson [2]. In recent years, it has received much attention due to the works of Melenk and Babuska [26] and Krysl et al. [27].

Shepard's approximation on a set of scattered points $\boldsymbol{x}$ of domain Ω is as follows:

$$u_h(\boldsymbol{x}) = \sum_{I=1}^{N} \omega_I(\boldsymbol{x})\, u_I ,$$

where u_I are the nodal parameters, and $\omega_I(\boldsymbol{x})$ are the basis functions of compact support. They are constructed from weight functions $W_I(\boldsymbol{x})$ by means of the formula

$$\omega_I(\boldsymbol{x}) = W_I(\boldsymbol{x}) / \sum_{k=1}^{N} W_k(\boldsymbol{x}).$$

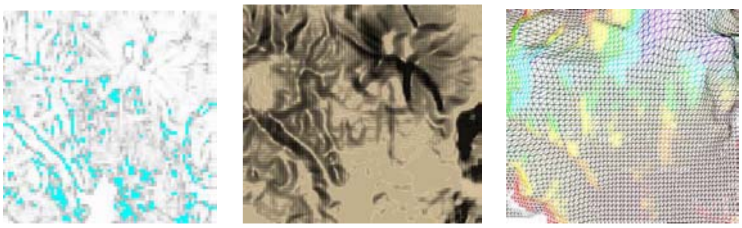
The CSRBF is used as a weight function

$$W_I(\boldsymbol{x}) = \begin{cases} (1-r)^4(4r+1), & 0 \le r \le 1 \\ 0, & r > 1 \end{cases},$$

where r = $\|\boldsymbol{x} - \boldsymbol{x}_I\|$ is the Euclidean distance between an interpolated point and an input point, and N is the number of points in a predefined area. Other choices of the weight function are also acceptable; however, theoretical proofs can be given to show that, to achieve extrapolation efficiency, weight functions, with small third derivatives should be used.

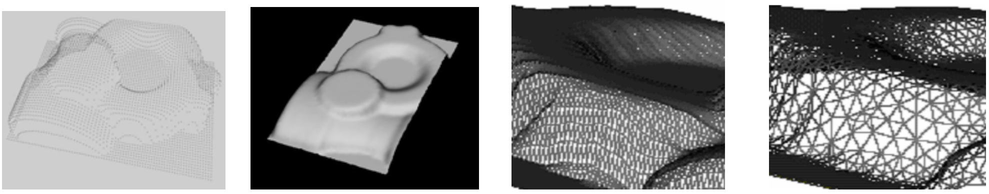
A general cover construction algorithm or partition of the domain Ω into overlapping rectangular patches ω_I to cover the complete domain has to be used. Let us note that our main premise is to take account of a surface variation σ (an analog of surface curvature) that might be useful for correct choice of a radius of support (r-sphere) for reconstruction taking into consideration the orientations of local surface elements. In our work [28], we have investigated rather simple scheme taking account of the local geometry of a surface.

Fig. 9 shows the results of reconstruction using the approach discussed above.



(a) (b) (c)

Figure 9: Implementation of the partition of unity for generation of polygons from scattered data of the fragment of Mount Bandai: (a) Curvature analysis. In the blue area, the surface variation $\sigma > 0.3$. (b) Result of reconstruction (ray tracing). Number of scattered points: 10000, processing time: 0.941 sec, number of vertices after reconstruction: 90000. (c) Fragment of the mesh as a wire-frame with color attributes according to calculated heights.



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

Figure 10: Surface reconstruction of a technical data set. (a) Cloud of points (4100 scattered points are used). (b) Simplified mesh shaded (processing time: 0.1 sec). (c) Fragment of the initial mesh, 31234 triangles., (d) Combined mesh modification (polygon reduction and statistical improvement of the mesh), 12132 triangles.

Fig. 10 shows an implementation of the PUM for generation of polygonal surfaces for point sets represented by elevation data. We demonstrate the applicability of the approach to data homeomorphic to a disc; nevertheless, since a closed object can be partitioned into a collection of bordered patches homeomorphic to a disc, this is no serious restriction, as it was mentioned by Horman and Greiner in [29].

3.6 Polygon simplification

Surface remeshing has become very important today for CAD and CG. This question is also very important for technologies related to engineering applications. Simplification of a geometric mesh involves constructing a mesh element which is optimized to improve the element's shape quality. Recently, a tremendous number of very sophisticated algorithms have been invented to obtain a simplified model. One exceedingly good overview [30] presents a problem statement and a survey of polygonal simplification methods and approaches.

We present here in more details a surface simplification method which uses predictor-corrector steps for predicting candidate collapsing points with consequent correcting them by the use of a statistical approach for triangles enhancement. The predictor step is based on the idea of selecting candidate points according to a bending energy.

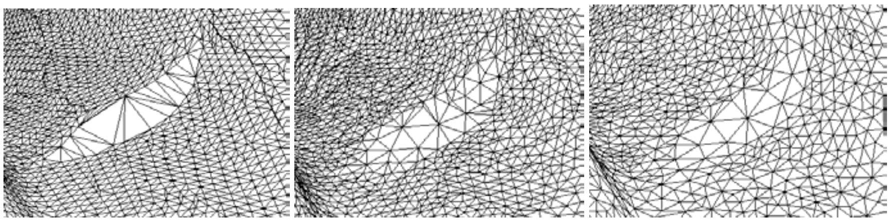
We show that since the simplification method is sufficiently efficient for up to 90 percent of reduction, there is no need for user-tuned parameters and the approach allows for obtaining a realistic time response (few seconds on AMD Athlon 1000 MHz) for sufficiently complex models (70K triangles).

Finding the optimal decimation sequence is a complex problem. The traditional strategy is to find a solution that is close to optimal; this is a greedy strategy, which involves finding the best choice among all candidates. Our simplification algorithm is sufficiently simple and is based on an iterative procedure for performing simplification operations. In each iteration step, candidate points for an edge collapse are defined according to a local decimation cost of points belonging to a shaped polygon. We call such a polygon a star. After all candidates have been selected, we produce a contraction step by choosing an optimal point.

A specific error metric is employed. We propose using the bending energy $\boldsymbol{h}^t\boldsymbol{A}^{-1}\boldsymbol{h}$ as an error/quality cost to select candidates for an edge collapse. The approach is based on the use of displacements of N control points as the difference between the initial and final geometric forms. The central point of a star polygon is considered as a point that can slide to the neighboring points. The selection of candidate points is made according to the bending energy. We exploit a simple idea that the more smoothly we transform a central point, the fewer residuals there will be between an initial mesh and the subsequent mesh. In this step, we form a list of points to be contracted; this list contains a number of candidate points. In the contraction step we eliminate processing of points that can be contracted twice or more.

Vertex placement is produced in two steps. In the first step we generate a position on the line connecting two vertices of an edge to be contracted. In fact, the optimal point is generated at the next step. The main underlying assumption of the algorithm is that a local mesh refinement automatically results in improved global mesh quality, bearing in mind the distribution of a limited set of polygons in the entire model.

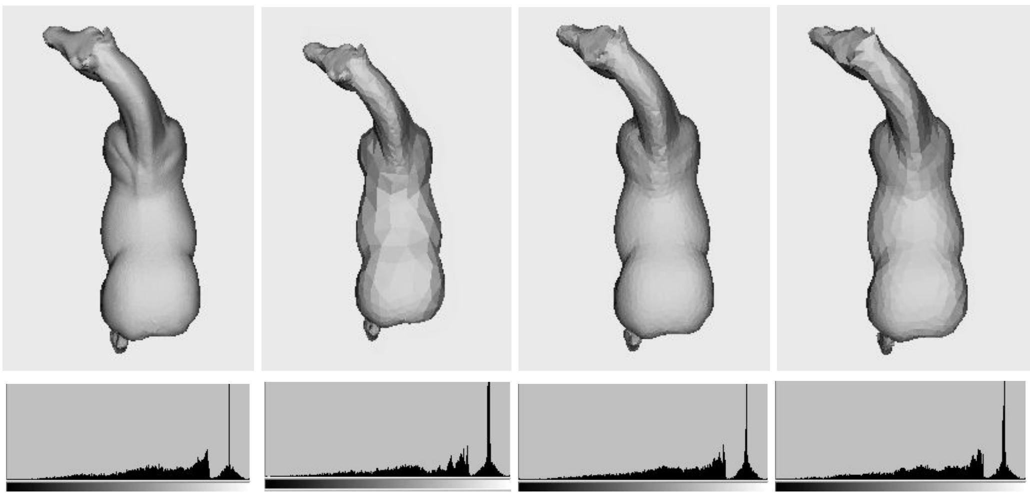
Statistics (see, [31]) on the values of the mesh quality criteria parameters of the neighbors of each vertex of the triangle mesh in order to predict the most likely value are exploited. This provides some latitude in the choice of point placement allowing softer "transformations" of polygons to be produced. Fig. 11 shows examples of polygon simplification of the "Horse" model.



(a) (b) (c)

Figure 11: (a) Fragment of the "Horse" model; (b) Fragment of the mesh after statistical processing; (c) Combined mesh modification: polygon reduction (40% of the original number of triangles) and statistical improvement.

Fig. 12 shows a good visual appearance of simplified models that is verified by luminosity histograms.



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

Figure 12: (a) Original "Horse" model (96966 triangles) and a luminosity histogram; (b) Simplified model produced accordingly to the use of the bending energy (10% of the original number of triangles) and the luminosity histogram; (c) Simplified model produced accordingly to the use of the bending energy and statistical approach (9% of the original number of triangles) and the luminosity histogram; (d) Simplified model produced accordingly to the use of the bending energy and statistical approach (3% of the original number of triangles) and the luminosity histogram.

Notice that in all the examples in this paper the processing time is shown for our test configuration AMD Athlon 1000 Mhz, 128 MB RAM, Microsoft® Windows 2000, ATI Radeon 8500 LE.

4. CONCLUSION

RBFs seem ready-made for many applications in shape modeling and CG, even for interactive 3D modification and sculpting. We have to state that according to our experiments with various applications of RBFs for surface modifications, for instance teeth reconstruction and optical design, we have a good alliance of geometric modeling and optimization techniques to determine the reconstructed surface and assure overall smoothness.

There is no single restoration and simplification method that provides the best results for every surface in the sense of quality and processing time. Experimental results indicate that the algorithms discussed in this paper provide rather good results and look promising for implementation in CAD and computer-aided engineering applications.

Let us note here that, for example shown in Fig. 10(d), the volume was well preserved, with a difference between the initial mesh and the processed one of about 0.65%.

One shortcoming of the approach discussed in section 3.5 is that in some areas (almost vertical) of a surface the triangular vertices may be spaced far apart. Selecting local data sets in the reconstruction algorithm according to the surface curvature is still an interesting research topic waiting for a solution. Our most urgent problem is to extend the algorithm given in section 3.5 to provide adaptive remeshing (enriching) according to local features of a surface geometry.

Algorithm proposed for hole filling based on polygonal stitching and space mapping techniques produces smooth and visually pleasant results, however, it is assumed that any hole does not have islands. In practice, holes filling problem becomes user dependent whether holes could be filled or not. Our current task is to find an approach to combine automatic defect detection and repairing algorithms.

Acknowledgments.

I am grateful to various people supported me to complete a piece of work presented here.

5. REFERENCES

[1] V. Savchenko and A. Pasko, "Transformation of Functionally Defined Shapes by Extended Space Mappings," *The Visual Computer*, 14, pp. 257-270, 1998.

[2] R. Franke and G. Nielson, "*Scattered Data Interpolation and Applications: A Tutorial and Survey, Geometric Modeling, Methods and Applications*", H. Hagen and D. Roller (Eds.), Springer-Verlag, pp. 131-160, 1991.

[3] V. Savchenko, V. Vishnjakov, "The Use of the "Serialization" Approach in the Design of Parallel Programs Exemplified by Problems in Applied Celestial Mechanics", *Proceedings of Performance Evaluation of Parallel Systems workshop, PEPS'93*, University of Warwick, Coventry, UK, 29-30 Nov., pp. 126-133, 1993.

[4] P. Litwinovicz, L. "Williams, Animating Images with Drawing," *Proceedings of SIGGRAPH`94,* pp. 409-412, 1994.

[[5] J. C. Carr, W. R. Fright and R. K. Beatson, "Surface Interpolation with Radial Basis Functions for Medical Imaging", *IEEE Transaction on Medical Imaging*, 16(1), pp. 96-107, 1997.

[6] F. L. Bookstein, "Principal Warps: Thin Plate Splines and the Decomposition of Deformations", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6), pp 567-585, 1989.

[7] V. Savchenko, A. Pasko, O. Okunev, and T. Kunii, "Function Representation of Solids Reconstructed from Scattered Surface Points And Contours", *Computer Graphics Forum, 14(4)*, pp. 181-188, 1995.

[8] R. K. Beatson and W. A. Light, "*Fast Evaluation of Radial Basis Functions: Methods for 2-D Polyharmonic Splines*", *Tech. Rep. 119*, Mathematics Department, Univ. of Canterbury, Christchurch, New Zealand, Dec. 1994.

[9] W. Light, "Using Radial Functions on Compact Domains", *Wavelets, Images and Surface Fitting*, P. J. Laurent et al. (Eds), AL Peters Ltd., pp. 351-370, 1994.

[10] B. Morse, T. S. Yoo, P. Rheingans, D. T. Chen, and K. R. Subramanian, "Interpolating Implicit Surfaces from Scattered Surface Data Using Compactly Supported Radial Basis Functions", *Proceedings of Shape Modeling conference, SMI`2001*,Genova, Italy, May, pp. 89-98, 2001.

[11] J. C. Carr, T.J. Mitchell, R. K. Beatson, J.B. Cherrie, W. R. Fright, B. C. McCallumm, and T. R. Evans, "Reconstruction and Representation of 3D Objects with Radial Basis Functions", *Proceedings of SIGGRAPH*, August, pp. 67-76, 2001.

[12] L. Greengard and V. Rokhlin, "A Fast Algorithm for Particle Simulation", *J. Comput. Phys*, 73, 325-348, 1997.

[13] V. A. Vasilenko, "*Spline-functions; Theory, Algorithms, Programs*," Novosibirsk, Nauka, 1983.

[14] H. Wendland, "Piecewise Polynomial, Positive Definite And Compactly Supported Radial Functions of Minimal Degree", *Adv. in Comput. Math.* 4, pp. 389-396, .1995.

[15] N. Kojekine, I. Hagiwara, and V. Savchenko, "Software Tools Using CSRBFs for Processing Scattered Data," *Computer & Graphics*, 27, pp. 311-319, 2003.

[16] Y. Ohtake, A. Belyev, and H-P. Seidel, "A Multi-scale Approach to 3d Scattered Data Interpolation with Compactly Supported Basis Functions," *In proceeding of SMI`2003*, Soul, South Korea, 12-16, May, 2003

[17] A. George and J. W. H. Liu, "*Computer Solution of Large Sparse Positive Definite Systems*," Englewood Clifts, NJ: Prentice-Hall, 1981.

[18] V. Savchenko and L. Schmitt, "Reconstructing Occlusal Surface of Teeth Using A Genetic Algorithm with Simulated Annealing Type Selection." *Proceedings of 6th ACM symposium on solid modeling and application*, Ann arbor, MI, USA, 4-8 June 2001, pp. 39-46, 2001.

[19] K. Myszkowski, V. V. Savchenko and T.L. Kunii, "Computer Modeling for The Occlusal Surface of Teeth", *Proceedings of CGI'96 Conf.*, Pohang, South Korea, June 24-28, 1996, pp. 196-198, 1996.

[20] N. Kozhekin, V. Savchenko, M. Senin, and I. Hagiwara, "An Approach to Surface Retouching and Mesh Smoothing," *The Visual Computer*, 19(7-8), Dec. 2003, 549-563

[21] A.N. Hirani, and T. Totsuka, "Combining Ffrequency and Spatial Domain Information for Fast Interactive Image Noise Removal", *Proceedings of SIGGRAPH`96*, pp. 269-276, 1996.

[22] G. Walberg, "Skeleton Based Image Warping," *The Visual Computer*, Vol. 5, No 1/2, March 1989, 95-108

[23] T. Beier and S Neely, "Feature-based Image Metamorphosis," *Computer Graphics*, 26,2, July 1992, pp. 35-42, 1992.

[24] N. Kojekine, V. Savchenko, M. Senin, and I. Hagiwara, "Real-time 3D Deformations by Means of Compactly Supported Radial Basis Functions," *Short paper proceedings of Eurographics`02*, Sarrbrucken, Germany, 2-6 September 2002, pp. 35-43, 2002.

[25] D. Shepard, "A Two-Dimensional Interpolation Function for Irregularly Spaced Data," *Proceedings of the 23th Nat. Conf. of the ACM* , pp. 517-523, 1968.

[26] J. M. Melenk and I. Babuska, *"The Partition of Unity Finite Element Method: Basic Theory and Applicationsd,"* Research Report No. 96-01, Seminar fur Angewandete Mathematik

Eidgenossische Technische Hochschule CH-8092, Zurich, Switzerland, 1996.

[27] P. Krysl and T. Belytchko, "An Efficient Linear-precision Partition of Unity Basis for Unstructured Meshless Methods," *Communications in Numerical Methods in Engineering*, 16, pp. 239-255, 2000.

[28] V. Savchenko, N. Kojekine, M. Savchenko, I. Semenova, O. Egorova, and I. Hagiwara, "Mesh Generation and Refinement of Polygonal Data Sets," *Proceedings of International Conference on CYBERWORLDS* (*CW 2003*), organized by: School of Computer Engineering, Nanyang Technological University in cooperation with: IEEE Computer Society, ACM SIGGRAPH (Singapore), and Eurographics, Singapore, December 3-5, 2003, pp. 198-205, 2003.

[29] K. Hormann and G. Greiner, "Quadrilateral Remeshing," *Proceedings of Vision, Modeling and Vizualization*, 2000, pp. 153-162, 2000.

[30] M. Garland, "A Multiresolution Modeling: Survey & Future Opportunities," *EUROGRAPHICS `99,* State of the Art Reports, 1999.

[31] O. Egorova, N. Kojekine, I. Hagiwara, M. Savchenko, I. Semenova, and V. Savchenko, "Improvement of Mesh Quality Using a Statistical Approach", *Proceedings of the 3rd IASTED International Conference on Visualization, Imaging and Image Processing*, Spain, 2003, Vol. 2, pp. 1016-1021, 2003

About the author

Vladimir Savchenko is a professor at Hosei University, Japan, Faculty of Computer and Information Sciences