Technical Report 2004-1-001

Shape recovery using functionally represented constructive models

Pierre-Alain Fayolle, Alexander Pasko, Elena Kartasheva, Nikolay Mirenkov

March 1, 2004

Software Department
The University of Aizu
Tsuruga, Ikki-Machi, Aizu-Wakamatsu City
Fukushima, 965-8580 Japan

Technical Report 2004-1-001

Title:

Shape recovery using functionally represented constructive models

Authors:

Pierre-Alain Fayolle*, Alexander Pasko**, Elena Kartasheva***, Nikolay Mirenkov*

Key Words and Phrases:

shape recovery, function representation, fitting, non-linear optimization, finite element meshes

Abstract:

We propose a method which helps to fit existing parameterized function representation (FRep) models to a given dataset of 3D surface points. Best fitted parameters of the model are obtained by using a hybrid algorithm combining simulated annealing and Levenberg-Marquardt methods. The efficiency of the approach is shown for recovery of three test items. We show through the CAD model processing an application of the proposed approach to the shape recovery followed by finite element mesh generation and adaptation.

Report Date:

3/1/2004

Written Language:

English

Any Other Identifying Information of this Report:

Distribution Statement:

First Issue: 1 copie

Supplementary Notes:

* The University of Aizu, Aizu Wakamatsu, Japan

** Hosei University, Tokyo, Japan

*** Institute for mathematical modeling, Russian Academy of Science, Moscow, Russia

Distributed and Parallel Proceesing Laboratory
The University of Aizu
Aizu-Wakamatsu
Fukushima 965-8580
Japan

1 Introduction

One of the actual problems in solid modeling is dealing with objects which are not yet (or not anymore) available as solid models. It is necessary to handle them in the same terms as other solid models. Therefore, special methods are needed for reverse engineering such solids. In this paper, we restrict our work to the shape recovery of constructive solids with smooth surfaces for cultural heritage and finite element meshes (FEM) applications.

One of the goals is to have models that can be later reused for modeling, modifications, or analysis. For instance, example-based modeling techniques are discussed for the case of human body in [20].

Previous work Reverse engineering of solid models relies on fitting some mathematical models, traditionally parametric or algebraic surfaces [1], [5], to scan data. Fitting a model consists in finding the best set of parameters such that the model becomes as close as possible to the data points.

Benko et al. [1] propose to fit multiple curves and surfaces to 2D and 3D real data, where the accuracy of the fit is enhanced by satisfying constraints between the parameters of the curves or surfaces. They introduce for that purpose an efficient constrained fitting scheme and derive a distance approximation for each primitive in their library for accurate and efficient fitting.

Fisher [5] proposes to use as much knowledge as possible to enhance quality of the reverse engineering process. Relations between parameters or objects are introduced to guarantee the production of sufficiently accurate models, which reproduce symetry and regularity. In [18], the same team proposes an evolutionary method based on GENOCOP III [14] to resolve efficiently the hard problem of non-linear constrained fitting. In [4] and [3], they justify the choice of Euclidean distance instead of algebraic distance or Taubin distance for fitting problem. Using Euclidean distance in the fitting process guarantees a better accuracy, correctness and robustness.

These interpretations of shape recovery well suit boundary representation with segmented point clouds. The main problem with this approach comes from the difficulty to extend the set of allowable shapes, because a corresponding segmentation would be required, which can be difficult or even impossible in the case of complex blends or sweeps. Furthermore, it may be difficult for the resulting model, generally available as a Brep, to be reused in extended modeling or analysis.

We use a different interpretation of shape recovery and a different model, the function representation of objects [16]. In our approach, standard shapes and relations are interpreted as primitives and operations of a constructive model. The input information provided by the user is a template (sketch) model, where the construction tree contains only specified operations and types of primitives while the parameter values of operations and primitives are not defined and are recovered by fitting. In general, FRep models are non-linearly parameterized and fitting them to 3D data should be done by non-linear least square minimization of a fit function, which indicates how close the model is to the 3D points.

2 Parameterized function representation

2.1 Constructive function representation modeling

The function representation (FRep) was introduced in [16] as a uniform representation of multidimensional geometric objects. An object (point set) in multidimensional space is defined by a single continuous real-valued function of point coordinates $f(\mathbf{x})$ which is evaluated at the given point by a procedure traversing a tree structure with primitives in the leaves and operations in the nodes of the tree. The points with $f(\mathbf{x}) \geq 0$ belong to the object, and the points with $f(\mathbf{x}) < 0$ are outside of the object. The geometric domain of FRep in 3D space includes solids with non-manifold boundaries and lower dimensional entities (surfaces, curves, points) defined by zero value of the function.

An FRep model can be built in a constructive way, with abstract parameters. The modification of these parameters can result in various different shapes, which can also be tuned to fit some modeling criteria. In the rest of the paper, the notation $f(\mathbf{x}, \mathbf{a})$ is used for a parameterized FRep, where $\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{R}^3$ is a point in the 3D space and $\mathbf{a} = (a_1, \ldots, a_m) \in \mathbb{R}^m$ is a vector of m parameters.

2.2 Origin of the template model

Template model can exist in specialized libraries for each application domain (mechanical design, human prosthesis design, and others) and may be reused, or need to be created by the user. In the latter case, a modeling work needs to be done by a designer. A parameterized model can be created using measurements or scans of a typical object. The model is required to keep basic ratios of the measured sample object and to proportionally change the dependent parameters according to introduced constraints. In case of scanned data available for a typical object, fitting of the template parameters can be also employed to establish basic ratios and constraints.

3 Fitting parameterized FRep: local approach

3.1 Fitting problem formulation

The problem is to recover a solid from a set of 3D points, $S = \{\mathbf{x_1}, \ldots, \mathbf{x_N}\}$, scattered on the surface of the object. Given S, the task is to find the best configuration for the set of parameters $\mathbf{a}^* = (a_1, \ldots, a_m)$ so that the parameterized FRep model $f(\mathbf{x}, \mathbf{a}^*)$ closely fits the data points. $f(\mathbf{x}, \mathbf{a})$ is an FRep model, made in a constructive way, which approximates the shape of the solid being reverse engineered. The vector of parameters $\mathbf{a}$ control the final shape of the solid and the best fitted parameters should give the closest possible model according to the information provided by S.

For computing how close a given point is to the surface of the solid with the current set of parameters, a fitness function is needed. The FRep model $f(\mathbf{x}, \mathbf{a})$ itself can serve for defining such an algebraic distance, because of the natural distance property of FRep models. The fit function becomes the square of the defining function values at all points (the surface of the solid being the set of points with zero function value):

$$error(\mathbf{a}) = \frac{1}{2} \sum_{i=1}^{N} f^2(\mathbf{x_i}, \mathbf{a}) \tag{1}$$

which can be also rewritten under the following form:

$$error(\mathbf{a}) = \frac{1}{2}\sum_{i=1}^{N} f_i^2(\mathbf{a}) = \frac{1}{2}\mathbf{f}^t(\mathbf{a})\mathbf{f}(\mathbf{a}) \tag{2}$$

where $\mathbf{f}(.)$ is the vector with $i\!-\!th$ component being $f_i(.)$ Now, we are searching for the vector of parameters $\mathbf{a}^*$ minimizing the fit function from equation 1, or 2. We consider firstly local methods for the minimization of functions with nonlinear parameters.

3.2 Local non-linear minimization of least squares

The best set of parameters $\mathbf{a}^*$ is found by minimization of the least-square error (equations 1, or 2), which is a non-linear function of the parameters $\mathbf{a}$. Traditional methods for solving such problems are Levenberg-Marquardt methods [15] or Quasi-Newton type methods [2]. Algorithms for both methods follow a common scheme: they proceed iteratively from an initial set of parameters and try to converge to a minimum in the parameter space. They are local methods only, because they strongly depend on the initial parameters, used for starting the algorithm.

Such algorithms search in each iteration for a privileged direction to go in the parameter space and for a step to move in that direction. Levenberg-Marquardt (LM) and Newton (N) algorithms differ in the selection of direction and in the ways of computing the step, more details can be read in [15] for the Levenberg-Marquardt method, and in [2] for the Quasi-Newton method.

3.3 Discussion of local methods

Unfortunately, methods such as LM and QN can in general guarantee only convergence to a local minimum. For some parameter spaces with complex topology, like for example, where multiple local minima exist, such methods are likely to be trapped into a local optimum and to stop there. Good initial parameters are very important, because they will determine to which minimum the algorithm may converge. Usually, if the parameters are not in the neighborhood of the global minimum, it is unlikely to converge to it, and the method instead will reach a local minimum.

It is possible with some further analysis of the model to have some additional information for getting better estimation of the starting parameters, unfortunately all the parameters can not be guessed that way.

4 Simulated annealing for fitting parameterized FRep models

We propose here to use a sampling algorithm (simulated annealing, SA) as an alternative method for fitting a parameterized FRep model. Techniques based on simulated annealing were proven to be efficient for solving global optimization problems with complex parameter space topology [13], and [11].

General idea When trying to minimize an objective function usually only downhill are accepted, but with this method some uphill may be accepted. This acceptance is made with a probability $p(t)$, which is initially close to 1, and then decreases to 0, when the temperature of the system reduces.

The temperature evolution also called tempreraturе schedule or cooling schedule (because the temperature only decreases) is the difficult part of the algorithm. The cooling should be not too quick neither to slow.

A thermodynamic analogy This method was inspired by the behaviour of some thermodynamical process. In a liquid at high temperature, the molecules move freely with respect to one another. When the liquid is cooled down, the mobility of the molecules decreases, and finally stops. If the cooling is not too quick, then the system will finish in a state of minimum energy.

The probability of accepting an uphill step is made by analogy with the Boltzmann law, which states that a system in thermal equilibrum has its energy distributed probabilistically among all different energy states. Even for a low temperature there is a chance for the system to be in a high energy, so that it can escape the local minimum energy and find a better one.

Application to numerical computation Metropolis [13] applied such results to numerical computation. The choice of a good cooling schedule is considered as the difficult part of the algorithm [8]; we use in our experiments an implementation based on the algorithm described in [7], with a linear cooling schedule.

Discussion Methods based on simulated annealing may be a good alternative to classical local methods like LM or QN in global optimum search among many local optima, but they are facing some major problem of efficiency: the objective function needs to be sampled a huge number of times before reaching convergence [9].

Because of that, simulated annealing seems to be a less attractive option for nonlinear fitting of FRep. A suitable solution would be to combine it with a local method, and to switch between the methods once the neighborhood of the global minimum was reached; we explore this potential solution in the next section.

5 Hybrid fitting method

In this section, we propose an approach combining local methods like QN or LM with sampling methods like simulated annealing. By doing so, we wish to have a method able to avoid local minima, but still fast enough. We propose for that to distinguish two phases in the fitting process as follows:

- At the beginning, a sampling method based on simulated annealing is used to get a first fit of the model,
- Then, it switches to a local method to extract the parameters that fit closely the model to the 3D point set.

The first step should give a configuration in the parameter space being in the vicinity of the global minimum and thus should help avoiding local minima. The second step should guarantee a faster convergence and should also converge to a better fitted model than using a sampling algorithm alone.

At the beginning of the fitting process, the input consists of a preliminary selected or specially built parameterized FRep model, a set of data 3D points, and an initial parameters estimation. The values of the initial parameters are not so crucial, because the sampling SA algorithm can escape local minima. When the parameters have geometric interpretation, it is also possible to guess initial values, even if not accurate at all. In the case when parameters have less obvious meaning, like coefficients in blending operations, it is more difficult to provide good initial estimation, therefore, some random initial values in the parameter space may be chosen.

The switch to a direct method is performed when there are some indications that an acceptable fit was reached: it can be when the least-square error is under some given threshold or when the fitted model looks close enough to the point cloud with a visual feedback to the user.

The goal of the two-phase method described above is to provide a fitting system for parameterized FRep models that does not require a good initial configuration, can escape local minima, and has an acceptable convergence rate. The next section presents the current implementation and some experiments with the developed software system.

6 Experiments

6.1 Test part

The first test part was modeled for testing purposes using HyperFun[1] a language and a set of software tools for FRep geometric modeling. The surface of the object was sampled to create the data set of 10714 3D points (Fig. 1).

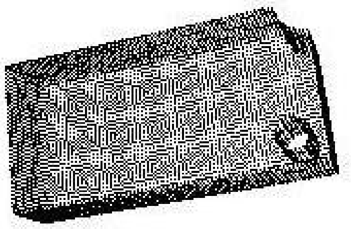

Figure 1: Data set used for the reverse engineering of the mechanical part. It contains 10714 3D points, scattered on the surface of the part.

The FRep defining function F shown below was used as a parameterized model for the recovery process:

$$\begin{aligned} f(\mathbf{x},\mathbf{a}) := (&box(\mathbf{x},\mathbf{a}) \backslash cylinderZ(\mathbf{x},\mathbf{a})) \\ &\backslash cylinderZ(\mathbf{x},\mathbf{a}); \end{aligned} \tag{3}$$

This FRep model consists of three simple primitives: one box and two infinite cylinders oriented along the Z axis, each primitive is defined by its parameterized model. For example, in the case of the cylinder, the defining function is: $cylinder(x,a) := a[1]^2 - (x[1]-a[2])^2 - (x[2]-a[3])^2$, where $a[1]$, $a[2]$, and $a[3]$ are parameters meaning the radius and a point on the $x-y$ plane, through which the axis of the cylinder passes. All the primitives are combined together using the R-Subtraction operator \, which is itself defined analytically as discussed in [19] and [16].

---

[1] www.hyperfun.org

set1	$\{-5, -4, -2, 10, 5, 4, 5, 3, 1, 3, -2, 1\}$
set2	$\{1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1\}$

Table 1: Two sets of initial values used for fitting: set1 is close to the best set of values, whereas set2 is completely wrong.

Twelve parameters were released in the model, which is the maximum number; they correspond to the lower-left corner of the box, three dimensions of the box, and the center and radius for each of the cylinders. The fitting algorithms need an initial estimation for the parameters, in the tests we used two sets of initial values configurations, they are given in the Table 1: one is close to the best fit (set1), while the second one is a wrong set (set2).

	Time in sec		Least square error	
	set1	set2	set1	set2
QN	1.852	9.643	5.47	595.04
SA	1635.09	1773.109	5.49	5.49
SAQN	72.042	144.177	5.47	5.47

Table 2: Time (in seconds) taken by each method to converge to the best fit and least-square error (sum of the deviations squared) of the best fit for each set of initial values.

The results of the tests are given in terms of the following: least square error of the reconstructed model for the three methods QN, SA, and hybrid SAQN (Table 2), time given in seconds taken to converge to the best fit for each of these methods (Table 2), and the visual shape of the best fit (Fig. 2).

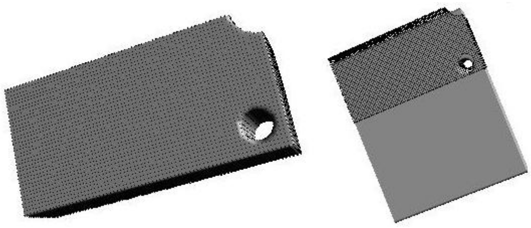

Figure 2: Shapes for the best fitted FRep in two cases: (right) best fitted, but wrong model when starting with set2 and using the QN method; (left) best fitted model when starting with the set2 and using the hybrid method.

Table 2 shows that the local method stops in a local minimum for the set2 of initial parameters, resulting in a wrong shape (Fig. 2, right shape), whereas with simulated annealing, it always converges to the global minimum. Unfortunately, the counterpart is the slow rate of convergence for the sampling method (Table 2). SA turns out to be 200 times slower.

When using a combination of SA and QN, we can always recover correct parameters and shape (Fig. 2, left shape, and Table 2, last line). The steps of the shape evolution during the hybrid method search are shown in Fig. 3. The overhead in speed is in magnitude of 10 compared to the local method (LN), but local optimum and bad shape recovery are avoided. Compared to SA, the hybrid method is between 10 and 20 times faster.

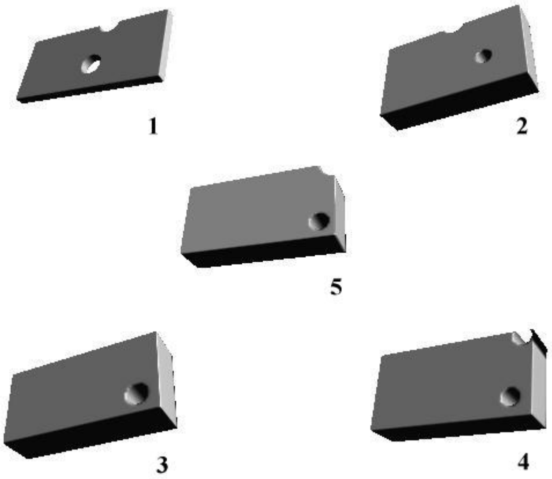

Figure 3: Evolution of the shape during the fitting process using the hybrid method.

6.2 Lacquer ware sake pot

The next example is the fitting of a model of a hand-crafted lacquer ware pot, which is used for pouring sake (Japanese rice wine). The discrete data set of the sake pot includes 27048 3D points, scattered on the surface of the object. The parameterized model of the sake pot sketched and discussed in the work on cultural heritage [17] is reused in our experiment. The parameterized model was created using hand measurements of a typical sake pot. The major parameters are the coordinates of the origin (position), the basic radius of the pot body, and the height of the pot handle. The model is required to keep basic ratios of the measured sample object and to proportionally change the dependent parameters like those of the blend area between the spout and the body, and the shape of the lid holder (note non-linear chages of these shapes in Fig. 5).

initial vector of parameters	0	0	5	1
final vector of parameters	2.705	0	4.804	7.71

Table 3: Initial and final vectors of parameters obtained by fitting with a local Newton method.

The first fitting test is made using a local Quasi-Newton algorithm. The results for the final vector of parameters are given in the Table 3. The value of the fit function (equation 1) is big enough to indicate that the method stopped at a local minimum. A comparison of the discrete model and the fitted parameterized FRep indicates clearly that the best fitted parameters corresponds to a local minimum of the fit function (see Fig. 4).

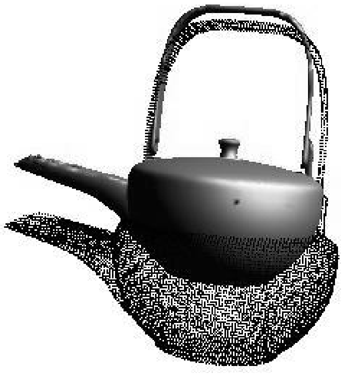

Figure 4: Local minimum effect: result of the fitting with a local method starting with wrong initial parameters. The discrete data points of the sake pot are also displayed for comparison.

A global fitting method needs therefore to be used in order to go to the vicinity of the global minimum of the fit function. The simulated annealing method is used next. The initial vector of parameters at the start of the process is the same as the one indicated in the Table 3 in the first line. The simulated annealing is stopped after the fit function value goes below a threshold given by the user. In the test, a value of 1000 for the fit function is used as a threshold value to determine the switch to a local method. This value corresponds to an average error of 0.04 of the fit function (Eq. 1), which we consider small enough in order to escape local minima and confirmed by visual feedback. This threshold is reached after 344 seconds on a Pentium IV PC. Then, the obtained parameters are reused as initial values for the local Quasi-Newton method. The steps of the evolution of the shape during the hybrid fitting of the FRep model can be seen at Fig. 5.

final vector of parameters	2.999	2.999	5.5	9.7044
fitting time	347 sec			
fit function	438.52			
mean error	0.03			

Table 4: Final vector of parameters after using the hybrid approach: a global minimization method (simulated annealing) followed by a local minimization method (Newton).

The final result for the best fitted parameters obtained after the two steps of the hybrid method is summarized in Table 4 with total time for the fitting, the fit function value and the mean error.

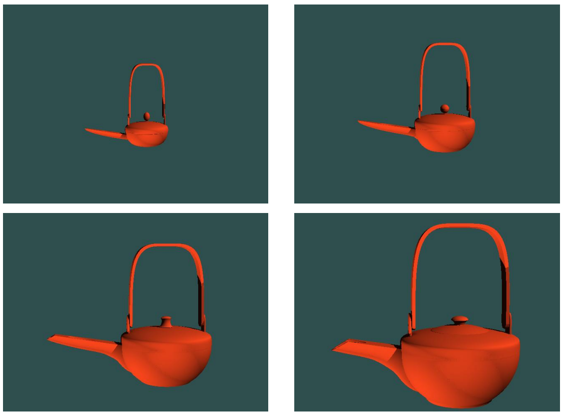

Figure 5: Evolution of the shapes of the sake pot during the fitting process

6.3 Application in Finite Element Meshes (FEM)

Approach to FEM generation Surface remeshing is very important for applications associated with numerical simulation procedures, in particular with finite element analysis (FEA). These applications impose strict constraints on the quality of the surface approximation and the shapes and sizes of mesh elements. Moreover finite element meshes have to be adapted both to physical and geometric features of computational tasks. Changes in the boundary or initial conditions of the simulated process may cause remeshing even if the computational domain remains the same. In many cases the initial description of computational domains in FEA is represented by their boundary surface triangulations. These triangulations can be exported from various modeling systems, produced by 3D scanning, or be a result of previous FE computations. Usually these initial triangulations consist of badly shaped triangles and are not adapted to physical conditions and an appropriate remeshing is required. Mesh refinement and optimization procedures need accurate information about the geometry of the computational domain. Therefore, the creation of an adequate description of a solid based on the initial triangulation of its boundary surface is an important problem for the FE mesh generation and optimization. Different approaches were considered to solve this problem. In [6], finite element adaptation is based on the local approximation of the underlying surface geometry by a quadric surface. The authors of [12] convert a CAD model into a volume representation by sampling its distance field on a uniform grid and then applying the extended marching cubes algorithm to this volume.

Taking into account that many mechanical parts can be represented as constructive solids we propose to apply FRep recovery to support FE mesh generation for objects whose initial geometry is represented by boundary surface triangulations. The initial mesh is used for the selection or creation of a parameterized FRep model. Then, the parameters of the FRep model are fitted to the vertices of the mesh. The final model can be used for the surface and volume finite element adaptation by the methods described in [10].

Fitting to a CAD mesh As an example of application of the FRep shape recovery for the FEM generation, a parameterized FRep model corresponding to the CAD mesh Fig. 6 (top, left) is created and fitted using the previously proposed techniques.

fit function	0.667
mean error	0.011622

Table 5: Fitting function value for the best fit set of parameters and the corresponding mean error.

The FRep model including 14 parameters is sketched corresponding to the shape shown in Fig. 6, top, left; the initial values for the parameters are chosen randomly.

The convergence is obtained using the method proposed in the paper, and the results in terms of the fitting function value and the mean error are presented in Table 5. The FRep shape corresponding to the best set of parameters is shown in Fig. 6, top, right.

Starting with the acquired FRep model, it is then possible to apply the mesh adaptation methods from [10]. The results of such methods are shown in Fig. 6, bottom. The left picture shows an optimized surface mesh, which was then used for the 3D tetrahedral mesh generation (right) using the extended advancing front method [10].

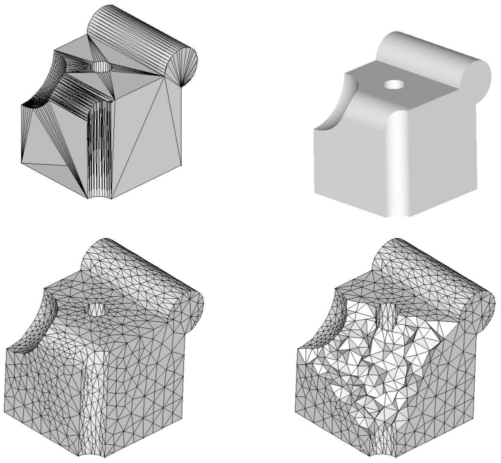

Figure 6: A surface mesh, generated by a CAD system (top, left), the recovered shape (top, right), the associated optimized mesh (bottom, left), and the 3D tetrahedral mesh generated from it (bottom, right).

7 Conclusion

We introduced an application of parameterized FRep models for shape recovery of constructive solid models from 3D point clouds. The use of parameterized FRep models presents some advantages over the traditional reverse engineering approach: no segmentation of the data set is required, a quite extensive and extensible set of FRep primitives and operations exist and can be utilized, all the primitives are defined by ready to use algebraic functions or procedures, and an existing special high-level language is available for the model representation.

The fitted template model can be analyzed, classified, or reused in other applications, for example, in further remodeling, animation, or rapid prototyping. Some of these applications may be more difficult or impossible to implement with Brep models, which are the results of traditional reverse engineering methods.

Nevertheless, the proposed approach revealed also some existing problems. A generic parameterized model used for the fitting has to be available or to be created by the user. The evaluation of the defining function for a complex shape can be also time-consuming, which results in a time consuming optimization process.

There are still ways to improve the presented approach. The semi-automatic creation of generic models can be envisaged with application of genetic algorithms and genetic programming. The problem of the parameterized model adequacy evaluation has to be seriously considered when using these methods. The global fitting method can also be enhanced by replacing simulated annealing with genetic algorithms.

Acknowledgements

P.-A. Fayolle acknowledges support by Monbusho, the Japanese Ministry of Education and Research.

References

[1] P. Benko, G. Kos, T. Varady, L. Andor, and R. Martin. Constrained fitting in reverse engineering. *Computer Aided Geometric Design*, 19:173–205, 2002.

[2] J. Dennis, D. Gay, and R. Welsch. An adaptative nonlinear least-squares algorithm. *ACM Transaction on mathematical software*, 7:348–368, 1981.

[3] P. Faber and R. Fisher. Euclidean fitting revisited. In *Proceedings of the 4th international workshop on visual form*, pages 165–175. Springer-Verlag, 2001.

[4] P. Faber and R. Fisher. Pros and cons of euclidean fitting. In *Proceedings of Annual German symposium for pattern recognition*, pages 414–420. Springer, Berlin, 2001.

[5] R. Fisher. Applying knowledge to reverse engineering problems. In *Proceedings of Geometric Modeling and Processing*, pages 149–155, 2002.

[6] P. Frey and H. Borouchaki. Geometric surface mesh optimization. *Computing and visualization in science*, 1(3):113–121, 1998.

[7] W. Goffe, G. Ferrier, and J. Rogers. Global optimization of statistical functions with simulated annealing. *Journal of econometric*, 60:65–99, 1994.

[8] B. Hajek. Cooling schedules for optimal annealing. *Mathematics of Operations Research*, 13:311–329, 1988.

[9] A. Hu, R. Shonkwiler, and M. Spruill. Estimating the convergence rate of a restarted search process. *International Journal of computational and Numerical analysis and Applications*, 1, 2002.

[10] E. Kartasheva, V. Adzhiev, A. Pasko, O. Fryazinov, and V. Gasilov. Discretization of functionally based heterogeneous objects. In G. Elber and V. Shapiro, editors, *ACM Symposium on solid modeling and applications*, pages 145–156. ACM, 2003.

[11] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[12] L. Kobbelt, M. Botsch, U. Schwanecke, and H.-P. Seidel. Feature sensitive surface extraction from volume data. In *Procedings of SIGGRAPH 2001*, pages 57–66. ACM, 2001.

[13] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equations of state calculations by fast computing machine. *J. Chem. Phys.*, 21:1087–1092, 1953.

[14] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, third edition, 1996.

[15] J. Moré. The Levenberg-Marquardt algorithm implementation and theory. In *Lecture notes in mathematics No630 Numerical analysis*, volume 630, pages 105–116. New-York, 1978.

[16] A. Pasko, V. Adzhiev, A. Sourin, and V. Savchenko. Function representation in geometric modeling: concepts, implementation and applications. *The Visual Computer*, 11:429–446, 1995.

[17] G. Pasko, A. Pasko, C. Vilbrandt, and T. Ikedo. Virtual Shikki and Sazaedo: shape modeling in digital preservation of japanese lacquer ware and temples. In R. Durikovic and S. Czanner, editors, *Spring Conference on Computer Graphics SCCG2001*, pages 147–154. IEEE, 2001.

[18] C. Robertson, R. Fisher, D. Corne, N. Werghi, and A. Ashbrook. An evolutionary approach to fitting constrained degenerate second order surfaces. In *Evolutionary Image Analysis, SIgnal Processing and Telecommunications. Proc. First European workshop on evolutionary computation in image analysis and signal processing*, pages 1–16. Springer LNCS, 1999.

[19] V. Rvachev. On the analytical description of some geometric objects. *Reports of Ukranian Academy of Sciences*, 153:765–767, 1963.

[20] H. Seo and N. Magnenat-Thalmann. An example based approach to human body manipulation. *Graphical Models*, 66(1):1–23, 2004.