Bayesian Mesh Reconstruction from Noisy Point Data

Guiping Qian, Ruofeng Tong, Wen Peng, and Jinxing Dong

State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou, 310027, China
qianguiping@163.com, trf@zju.edu.cn, pengwen@zju.edu.cn,
djx@cs.zju.edu.cn

Abstract. This paper presents a new reverse engineering method for creating 3D mesh models, which approximates unorganized noisy point data without orientation information. The main idea of the method is based on statistics Bayesian model. Firstly the feature enhancing prior probabilities over 3D data retrieve the sharp features, such as edges or corners. Then a local polynomial probabilistic model is used to approximate a continuous differentiable manifold. The Bayesian model uses an iterative fitting clustering algorithm to improve the noise tolerance in geometry accuracy. After iteration a density-based estimation function can automatically remove the outliers. Furthermore, the current sphere cover meshing approach is improved to reconstruct the mesh surface from the noisy point data. Experimental results indicate that our approach is robust and efficient. It can be well applied to smoothing noisy data, removing outliers, enhancing features and mesh reconstruction.

Keywords: Mesh reconstruction, Bayesian model, reverse engineering, denoising.

1 Introduction

Surface mesh reconstruction from an unorganized point data has been a very important task and attracted increasing attention in various computer sciences, such as computer vision, virtual reality and CAD/CAM. Many 3D scanning devices can yield rather dense and accurate surface data samples. However, noise, outliers and defective data are inevitable because of the device precision and electronic disturbance. With these point cloud, a number of surface reconstruction techniques have been proposed recently. Despite of the versatility of large amount of algorithms nowadays, however, most of these algorithms are not efficiently enough and make certain strong assumptions on the original surface and its sample points. For example, some approaches become expensive in time and storage, many algorithms [2], [3], [5], [12] need additional knowledge such as surface normal or interior/exterior information, some are not tolerant of noise and defective data.

1.1 Previous Work

Statistical learning techniques, such as Bayesian method, support vector machines and clustering, have begun to be applied to computer graphics during recent years. Statistical learning methods for denoising and retrieving sharp features have attracted

Z. Pan et al. (Eds.): ICAT 2006, LNCS 4282, pp. 819 – 829, 2006.

many researchers' interest. Diebel et al. [1] applied Bayesian model to probable surface reconstruction and decimation from noisy surface meshes. Pauly et al. [4] introduced the clustering method in shaping point cloud. More recently, mean shift method was applied to denoising of noisy point cloud by Schall et al. [7]. However, these two methods cannot retrieve sharp features. Robust statistics [11], such as least median of squares, also has obtained a good experimental effect on reconstructing a piecewise smooth surface, but it is low efficient.

The topic of meshing the scattered point data with sharp features has a long history. Hoppe et al. [13] used a sign distance function to handle different topologies, and they approximate the normal at each sample using principal component analysis, which had been adopted by many researchers. Recently Ohtake et al.[5], Kobbelt et al.[12] extracted the edges or corners by the intersection of two or more piecewise quadratic functions. Both of these methods need the exterior additive accurate normal information. In [6], Ohtake used a local quadric error minimization strategy to get sharp features, but this method may fail to afford a satisfactory reconstruction when the level of noise is very high. Newly a particularly powerful point set surface approach, the moving least squares (MLS), has attracted many researchers' interest [4], [8], [9], [14], [15]. These methods based on MLS and extremal surfaces are very difficult to obtain the sharp feature.

1.2 Contribution

The main contribution of our approach is to provide a very efficient, robust and satisfied mesh reconstruction from noisy point data. Compared to robust filtering of noisy scattered point data [7], our method can retrieve the sharp feature very efficient, and avoid the artificial selection of kernel size. Our approach is also closely related to the Bayesian method of mesh surface reconstruction [1], but our approach is specially designed to work on meshing the noisy scanning point data, and it need automatically remove the outliers.

The main idea of our approach can be stated as followed three steps:

Firstly a nonparametric kernel density function is used to relate the computed qualities to the sample density. It will afford reliable outlier removal and the weight of the point data. We make use of the Shannon entropy to optimize window-width parameter, which avoids the artificial selection of input parameter. The details are described in section 2.

Then a Bayesian probability is introduced to construct the surface model of noisy point data without orientation (explained in section 3). We use an iterative clustering algorithm, which is integrated with Bayesian model, to improve the noise tolerance in geometry accuracy. At the same time, the sharp feature is preserved and enhanced. After iteration, the detached outliers are removed by density function. The iteration process is explained in section 4.

Finally the new point set is generated. We mesh the new point set. The non-manifold mesh is refined by normal comparison between the new vertex and the mesh topology (described in section 5).

The main stages of our approach are described below in detail.

2 Kernel Density Function

Firstly we introduce some basic concepts and definitions. Let R^3 denote the 3D Euclidean space. Considering a set of scattered points $Ps=\{p_k\}_{k\in index(Ps)}$, $p_k\in R^3$, p_k is an arbitrary point in a point set. *index(Ps)* is the index of the point set. $|Ps|$ represents the total element numbers of point set *Ps*. To describe the local approximation error and weight function, we introduce the support set $SPT(p_k)\subset Ps$ and the support radius R_k of point p_k. Here, $SPT(p_k)=\{p_j\}$, $j\neq k$, p_j satisfies the expression $\| p_k-p_j \|< R_k$. $\|\bullet\|$ is the Euclidean distance in R^3.

2.1 Density Function

It is often important to attach the computed qualities to sample density for scanned scattered point set. Informally, the kernel density function is a mathematical description of the influence a point data has within its neighborhood. The density function at a point $x\in R^3$ is defined as the sum of the influence functions of all neighboring data at that point. Thus a kernel density function $Ds(x)$ can be given by

$$Ds(x)=\frac{1}{n}\sum_{i=1}^{n} e^{-\frac{\|x-p_i\|^2}{2\sigma^2}}. \tag{1}$$

The Smooth parameter σ is called window-width size. Like the clustering method [16], it is a local density function approximating the overall density function. The density threshold ξ determines our selection to outliers.

2.2 Optimized Parameter

The window-width parameter σ of $Ds(x)$ decides the influence field of each point in its neighborhood. Density-based method is an efficient clustering method, so it can discover outliers and be insensitive to noisy point data. However, this method badly depends on the selection of the parameter. We know that Shannon' entropy can be used to measure the uncertainty of system. In our density function, if function value distributes unevenly, the uncertainty will be small. To obtain an optimized density function, we introduce the density entropy:

$$En(Ps)=-\sum_{k=1}^{N}\frac{Ds(p_k)}{\sum_k Ds(p_k)}\log\frac{Ds(p_k)}{\sum_k Ds(p_k)}. \tag{2}$$

When $\sigma\to 0$, the density function value of each point approximate $1/n$ and the density entropy arrives at maximum $\log(n)$. When σ varys from zero to infinity, density entropy gradually decreases to some optimized value, whereafter increases to maximum again. Simply minimizing the density entropy $En(Ps)$ we obtain the optimized window-width σ. This method helps us to avoid the artificial repetitious selection of input parameter.

After shrinking iteration of our reconstruction model (in section 4), some outliers cannot move to the Bayesian probability surface. We found out that setting density

threshold $\xi = 0.4\overline{Ds}(x)$ is appropriate. $\overline{Ds}(x)$ is the mean density function value when point data even distributes. When $Ds(p_k) > \xi$, the point p_k is preserved for further process. Otherwise, we will regard it as outlier and remove it from point set *Ps* (see Fig. 1).

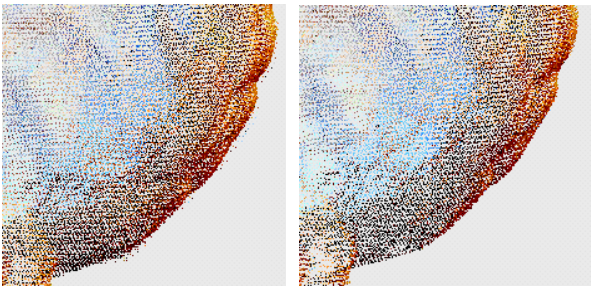

Fig. 1. Density function filtering. Left is the bosom of the raw Stanford bunny. Right is the result of density function filter the noisy point data. The outliers have been removed well.

2.3 Curvature Penalty

Weight is frequently used to give the different feature point a special influence. In this part, we will consider assigning an optimized weight to each noisy point data. Firstly we assess the normals through the correlation matrix M_k of neighboring points.

$$M_k = \sum_{p_i \in SPT(q_k)} (p_i - q_k)(p_i - q_k)^t . \tag{3}$$

Here q_k is the center of the neighborhood p_i. Its three eigenvalues are $\{\lambda_1, \lambda_2, \lambda_3\}$, $\lambda_1<\lambda_2<\lambda_3$. Their corresponding eigenvectors $\{v_1, v_2, v_3\}$ form an orthogonal basis and v_1 approximates the surface normal at q_k. We note the normal n_k of each point q_k. From [17], we know the relations of the eigenvalues and shape of correlation ellipsoid. We give each point a curvature penalty weight:

$$Cur(p_k) = (\lambda_2 - \lambda_1) / \lambda_3 . \tag{4}$$

In the presence of noise or near a discontinuity, the curvature penalty weight will be very small. Finally in our reconstruction model, we attach a weight w_k to each point p_k:

$$w_k = Ds(p_k) \cdot Cur(p_k) . \tag{5}$$

w_k will be large when the small neighborhood of a point is dense and flat (or local smooth). When the small neighborhood is noisy or local discontinuous, w_k is very small even approximating zero.

3 Reconstruction Model

We define a surface approximating the noisy point data as a differentiable manifold $S(x,y)$. Let x be the real surface sample point of the object being scanned. Instead of x, the sensor detects noisy measurements $\{y_i\}_{i=1,2,\ldots,N}$ of the surface of the object. The measurements are the noisy point data for our analysis. Intuitively, if we know some hints about the most probable location of the differentiable manifold in the 3D space, we can see this idea results from the probability density $P(x|\{y_i\})$. So we give the higher probability to the measurements which close to the most probable proximity of the manifold in space. Bayesian formula enables us to realize this probability.

3.1 Bayesian Probability

Given the measurements $\{y_i\}_{i=1,2,\ldots,N}$, with Bayesian rule we construct followed model

$$P(x \mid \{y_i\}) = \frac{P(\{y_i\} \mid x)P(x)}{P(\{y_i\})}. \tag{6}$$

Here $P(\{y_i\}|x)$ is the probability of the measurements in the hypothesis that y_i approximates x, and $P(x)$ is a prior probability distribution on surface of the object. From the maximum principle, the process of most probable surface reconstruction is to find the surface x that maximizes the posterior probability $P(x|\{y_i\})$ in (6). Because $P(\{y_i\})$ is a constant which is independent of x, maximizing $P(x|\{y_i\})$ just is maximizing $P(x)$ and $P(\{y_i\}|x)$ respectively. These two different parts need us to construct corresponding models. In the followed two paragraphs, we will present how to construct the prior probability $P(x)$ and measurements probability $P(\{y_i\}|x)$.

3.2 Prior Probable Surface

Without a prior, the most probable surface should be the measurement $\{y_i\}$ itself. The prior probable surface is quite important to our surface mesh reconstruction. Applying the prior probability distribution can help us to preserve or enhance the shape features.

We define the prior probable function

$$\varphi_k = \sum_{i \in SPT(p_k)} (w_i n_i \cdot (p_i - p_k))^2. \tag{7}$$

where weight w_i has been explained at paragraph 2. Thus each φ_k is a field potential and nonnegative. If all p_i and p_k are on the plane $C \subset R^3$, then $\varphi_k=0$. This field distance function is essential used to enhance the shape features, like edges or corners. It is developed firstly by Hoppe et al. [13]. From equation (7), we can see the prior probable function φ_k is linear piecewise, but the general fit functions, like piecewise quadratic polynomial, are wholly differential manifold, i.e. smooth and continuous.

When x locates on the prior surface function, φ_k reaches minimum. Then we think that surface prior probability has been maximized. The surface prior probability has followed formula

$$P(x) = -\ln(\varphi_x). \tag{8}$$

Negative natural logarithm is made the most of to obtain positive $P(x)$.

3.3 Manifold Surface

$P(\{y_i\}|x)$ has been defined for the probability of the measurements ahead. Here x is the true surface point of the scanned object. Subjected to sensor noise and merged error, the original scattered point cloud in hand are often very difficult to model directly. We can't assume that the preprocessing points follow the Gaussian probability distribution, because the point cloud is commonly merged by several layers. Thus such distribution probability is very difficult to formulate. Approximating the intrinsic surface of unorganized space points is still a quite young topic.

In defining point-set surfaces [14], [15], we know that the extremal surfaces, include MLS surface, are defined by an energy function and a vector field. The extremal surface is essential a smooth continuous manifold reconstruction. When multiple surfaces intersect, forming sharp feature, the vector field singularities will cause non-manifold singularities in the extremal surface. The non-manifold singularities just can be extracted by the prior probable function φ_k.

Inspired by point-set surface [15], we define an implicit function:

$$\phi(x) = n(x) \cdot \nabla f(x). \tag{9}$$

where $n(x)$ is vector field, and $\nabla f(x)$ is gradient potential field of iso-surface $f(x)$. $\phi(x)$ is a extremal manifold surface. The effect of $\nabla f(x)$ in our approach is that we can let the scattered point iteratively shrink towards the zero iso-surface along the vector field. Transforming equation (9) adapted to scattered points, we can obtain a more computable formation:

$$\phi(x) = \varepsilon \cdot n(x) \cdot (x - \frac{\sum_k w_k p_k}{\sum_k w_k}). \tag{10}$$

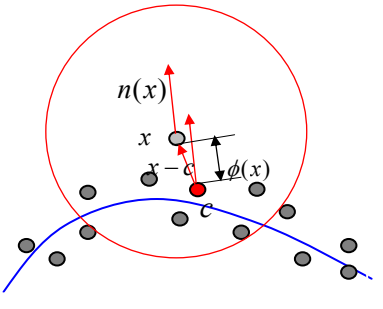

Fig. 2. The manifold surface function (a 2D example). Red vector x-c is used to approximate $\nabla f(x)$, when x locates on the blue line(manifold surface), $\phi(x)$=0. $\phi(x)$ is the approximative estimation of the distance between point x and manifold surface.

$x - \sum_k w_k p_k \Big/ \sum_k w_k$ is used to approximate $\nabla f(x)$. ε is a ratio coefficient and set 0.4~0.6 (see Fig. 2). If ε is too large, it will cause shape shrinking at high curvature regions. $\phi(x)$ might be negative, it isn't suitable for probabilistic measurement function. Easily we square $\phi(x)$ and use negative nature logarithm to generate probabilistic measurements function:

$$P(\{y_i\} \mid x) = -\ln(\phi(x)^2). \tag{11}$$

4 Shrinking Iteration

With equation (6) constructed, maximizing the posterior probability can obtain the optimized surface reconstruction. We invert equation (6) into:

$$\frac{\partial P(x \mid \{y_i\})}{\partial x} = \frac{1}{P(\{y_i\})} \cdot (P(x) \frac{\partial P(\{y_i\} \mid x)}{\partial x} + P(\{y_i\} \mid x) \frac{\partial P(x)}{\partial x}). \tag{12}$$

When equation (12) approximates zero, the posterior probability arrives at the maximum. Equation (12) can be decomposed into two key parts:

$$\frac{\partial P(\{y_i\} \mid x)}{\partial x} = 2\varepsilon^2 [n(x) \cdot (x - \frac{\sum_k w_k p_k}{\sum_k w_k})] n(x) \cdot \frac{1}{\phi(x)^2}, \tag{13}$$

$$\frac{\partial P(x)}{\partial x} = \sum_{i \in SPT(p_k)} 2w_i^2 [n_i \cdot (p_i - p_k)] n_i \cdot \frac{1}{\varphi_x}. \tag{14}$$

This means to move all measurements to positions of high probability. The iterative moving process alters the sample positions along the gradient ascent maximization. When equation (12) iteratively converges approximately to zero, we stop the iteration. To control the total computation time, In fact, we also need to decide reasonable steps. We use followed practical point-iteration equation:

$$p_i^{k+1} = p_i^k - \frac{\partial P(\{y_i\} \mid x)}{\partial x} \cdot \phi(x)^2 - \frac{\partial P(x)}{\partial x} \cdot \frac{1}{\sum_i w_i^2} \cdot \varphi_x. \tag{15}$$

When $\| p_i^{k+1} - p_i^k \| < 10^{-2}\sigma$, the iterations stop.

5 Meshing and Visualization

In the section 4, we have finished the shrinking iteration, and then we remove the detached outliers to acquire new point set. We need to connect them to generate mesh data struct Ω. The traditional meshing method is to compute the Delaunay triangulation. Computing the Delaunay triangulation can be slow and susceptible to numerical errors. Our meshing approach is based on spherical cover method [6].

We use equation (7) as an error function instead of Ohtake' local quadric error function. Then we generate the adaptive spherical cover. By this means, the resulting triangle mesh may have non-manifold parts and hole boundary. Each vertex with 1-ring neighborhood is classified: simple, non-manifold and hole boundary. We need to refine the original mesh further.

The optimal mesh should satisfied two conditions: the triangles are approximate to the surface of the object; the mesh is fair. Ohtake's mesh cleaning method takes the disk-shaped 1-ring neighborhood that has minimal curvature, which cannot satisfy the two conditions. By comparing the normal deviation of the neighboring triangles with the vertex, we remove disk-shaped 1-ring neighborhoods which have large normal deviation. From equation (3) we get the normal of the vertex $n(i)$. We define the normal of the kth disk-shaped 1-ring neighborhood of vertex **i**

$$N(k) = \sum \theta_j \xi_j \Big/ \sum \theta_j \,. \tag{16}$$

Where θ_j is the area of the triangle T_j, ξ_j is the normal of the triangle T_j. The deviation of the two normals is

$$D = \mid n(i) \cdot N(k) \mid. \tag{17}$$

If $D < 0.8$, the kth 1-ring neighborhood of vertex **i** is recognized as false and eliminated. If $D \geq 0.8$, then we compare the fairness of the triangle mesh. We define the fair parameter

$$\varepsilon(k) = \sum \ell_j / \theta_j \Big/ \sum 1/\theta_j \,. \tag{18}$$

Where ℓ_j is the length ratio of the longest edge and the shortest edge in triangle T_j. To fair the mesh we take the minimal $\varepsilon(k)$. So we get the optimal disk-shaped 1-ring neighborhood.

By above process, the hole boundary of the mesh is remained. We remove the isolated vertices and check the hole boundary loops to fill the holes. If the holes are too big, The subdivision can be adopted to increase the hole mesh density.

6 Results and Discussion

The goal of our work is to design an efficient and robust algorithm for meshing noisy point data. Our method can be applied to unorganized point set consisting of millions of points without orient information on standard PCs. We will show several representative experimental results for the Bayesian mesh model reconstruction. All examples presented in this paper are computed on a P4 2.0 GHz PC with 512Mb RAM running on WINDOWS 2000.

Next we will compare our mesh reconstruction method to the other method in existing literature. Fig. 3 compares Bayesian mesh reconstruction to the quadric-error function iteration method of Ohtake et al. [6]. Here we don't use the density function to filter the outliers. Obviously our method is very robust to smooth noise and recover the shape of the object. Fig. 4 shows the different mesh reconstruction results from

the Bimba model. The triangle faces numbers of two different Bimba mesh models are both 147K. These comparisons show our Bayesian probable surface mesh method performs well.

Table 1 shows the computational time comparison of Bayesian probable surface mesh reconstruction process for different models. However, dragon scans model with fewer points than Bimba scans consumes more time, because the point data of dragon scans is noisier and distributes unevenly. From the results we notice that out method is very fast. The memory usage of our method is very low. The peak RAM is 350Mb in processing AIM@SHAPE Bimba model with 1.8M scanning points.

Table 1. The experimental results on computational time and the size of the point set for different models

Point Set	Points N	Triangle F	Time
Bunny	361K	74.1K	17.5sec
scans	361K	185K	24.7sec
Dragon	1406K	21K	114.3sec
scans	1406K	243K	227.6sec
Bimba	1873K	149K	75.6sec
scans	1873K	255K	84.7sec

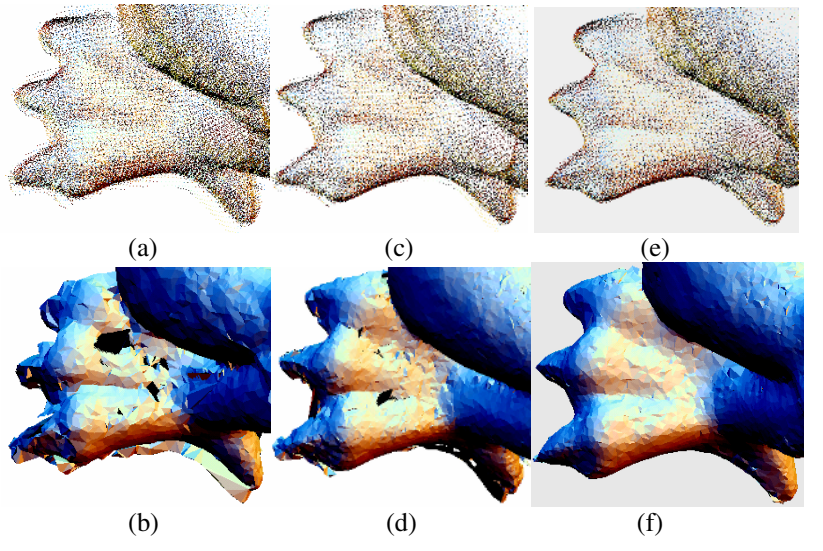

Fig. 3. A comparison to the public results of Ohtake et. al [6]. (a) (b) are raw scanned point data of Stanford dragon nail consists of 47 scans and mesh reconstruction [6]. (c) shows Ohtake's quadric-error iteration which is used to smooth noisy data in (a). (d) is mesh reconstruction. (e) shows the results of shrinking iteration of Bayesian probable surface in (a) and we don't remove the outliers. (f) is the mesh reconstruction of point data in (e).

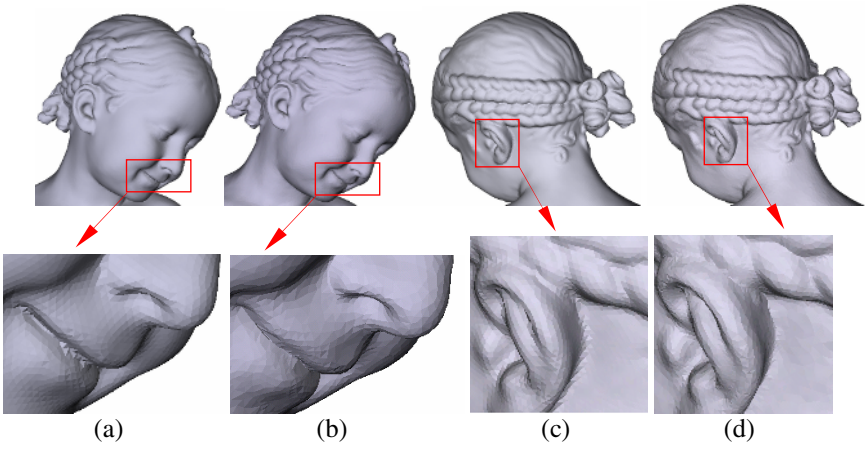

Fig. 4. A comparison to the public results of ReMESH [18]. (a) and (c) shows Simplification of scanned Bimba model with PlyDeci. Cleaning, hole-filling and smoothing with ReMESH [19]. (b) and (d) shows our results of Bayesian probable surface method. From the comparison our method can preserve and enhance the features.

7 Conclusion and Future Research

In this paper we present a novel noise-tolerable surface mesh reconstruction method, which is based on a statistics Bayesian model. Compared to other mesh reconstruction approach for processing noisy point data, our algorithm is very robust, efficient and easily to realized. Our method also consumes low memory with large point data set. The feature is enhanced by our prior probable surface model. We believe that our method can be applied to many surface reconstructions, such as multi-level smoothing, feature enhancing, noise filtering, and decimation.

Of course, the denoising procedure and automatic outliers removal remain an intensive research topic. Our method is an important step to the robust smooth, but so far no algorithms is perfect and satisfied to any noisy point data set. Many scanning model need new algorithm to apply. Our holes repair region may not look optimal when there are complex holes with highly curved shapes.

Acknowledgments. Project supported by the National Grand Fundamental Research 973 (No.2002CB312106) of China. The Bimba model is courtesy of AIM@SHAPE Shape Repository. The Bunny and Dragon models are courtesy of Stanford Computer Graphics Laboratory.

References

1. Diebel, J., Thrun, S., Bruning, M.: A Bayesian Method for Probable Surface Reconstruction and Decimation. ACM Transactions on Graphics, Vol. V, No. N, September, (2005) 1–20
2. Steinke, F., Schölkopf, B., Blanz,V.: Support Vector Machines for 3D Shape Processing. Computer Graphics Forum (Proc. EUROGRAPHICS) 24(3), (2005) 285 – 294

3. Schölkopf, B., Steinke, F., Blanz, V.: Object Correspondence as A Machine Learning Problem. Proceedings of the 22nd International Conference on Machine Learning,(2005) 777 - 784
4. Pauly, M., Gross, M., Kobbelt, L. P.: Efficient Simplification of Point-sampled Surfaces. IEEE Visualization, (2002) 163– 170
5. Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., Seidel, H.-P.: Multi-level Partition of Unity Implicits. ACM Transactions on Graphics 22, 3 (July), page 463-470, Proceedings of SIGGRAPH 2003
6. Ohtake, Y., Belyaev, A., Seidel, H.-P.: An Integrating Approach to Meshing Scattered Point Data. ACM Symposium on Solid and Physical Modeling, (2005)
7. Schall, O., Belyaev, A., Seidel, H.-P.: Robust Filtering of Noisy Scattered Point Data. Eurographics symposium on point-based graphics, (2005)
8. Adamson, A., Alexa, M.: Approximating and Intersecting Surfaces from Points. In Proceedings of EG Symposium on Geometry Processing, (2003) 245–254
9. Amenta, N., Kil, Y. J.: Defining Point Set Surfaces. In Proceedings of SIGGRAPH, (2004) 264-270
10. Ju, T., Losasso, F., Schaefer, S., Warren, J.: Dual Contouring of Hermite Data. ACM Transactions on Graphics, 21, 3, (2002) page 339-346
11. Fleishman, S., Cohen-or, D., Silva, C.T.: Robust Moving Least-squares Fitting with Sharp Features. In Proceedings of ACM SIGGRAPH, (2005)
12. Kobbelt, L.P., Botsch, M., Schwanecke, U., Seidel, H.-P.: Feature Sensitive Surface Extraction from Volume Data, In Proceedings of ACM SIGGRAPH, (2001) 57–66.
13. Hoppe, H., Derose, T., Duchamp, T., Mcdonald, J., Stuetzle, W.: Surface Reconstruction from Unorganized Points. In Proceedings of ACM SIGGRAPH, (1992) 71-78
14. Amenta, N., Kil, Y.J.: The Domain of A Point-set Surface. Eurographics Workshop on Point-based Graphics, (2004) 139-147
15. Amenta, N., Kil, Y.J.: Defining Point Set Surfaces. In Proceedings of SIGGRAPH, (2004) 264-270
16. Hinneburg, A., Keim, D. A.: An Efficient Approach to Clustering in Large Multimedia Databases with Noise, In Proceedings of the 4th International Conference on Knowledge Discovery and Data mining, (1998) 58-65
17. Gumhold, S., Wang, X., Mcleod, R.: Feature Extraction from Point Clouds. In Proceedings of 10th International Meshing Roundtable, Sandia National Laboratories, (2001) 293-305
18. Attene, M., and Falcidieno, B.: ReMESH: An Interactive Environment to Edit An Repair Triangle Meshes, SMI 2006
19. http:// shapes.aim-at-shape.net/viewmodels.php?page=2