



Enhanced Surface Reconstruction Algorithm Using Umbrella Filtering

Abhishek Bansal
Department of CSE,
Punjab Technical University

Vishal Arora
Department of CSE,
Punjab Technical University

Niraj Madaan
Department of CSE,
Kurukshetra University

Abstract— the goal of surface reconstruction is to find a surface from a given finite set of geometric sample values. In many applications, the sample values are points. But other types of samples, like curves occurring e.g. in tactile sampling by an adapted milling machine, or volume densities occurring for instance in X-ray based computer tomography, are also possible. Reverse engineering of geometric shapes is the process of converting a large number of measured data points into a concise and consistent computer representation. In this sense, it is the inverse of the traditional CAD/CAM procedures, which create physical objects from CAD models.

Triangulating scattered point-sets is a very important problem of reverse engineering. Given a set of unorganized points that lie approximately on the boundary surface of a three-dimensional object, and without a priori information on the topology, our goal is to reconstruct the surface by building a triangular mesh using the given points as vertices. The resulting polyhedron can be the input of further procedures like surface fitting, or can be visualized with various textures. (For example, in computer-animated films the characters are often created as clay models first, then the 3D scanned and triangulated models are used for visualization.)

Keywords—CAD, ICADM, surface reconstruction, triangulated models are used for visualization, database, query processing.

I. INTRODUCTION

The problem of creating a CAD model for an existing physical object from a given set of points of the object surface is important in many fields of science and industry. There are many methods available for solution of this problem. These methods are based on a great variety of principles, and have various properties, that in many cases allows choosing the most suitable algorithm for a given task. At the same time many authors don't consider two problems, which often arise in practice. The first one is that we often deal with very large clouds of points (for example, clouds representing buildings, big sculptures, landscape areas, etc.). Processing such clouds of points often leads to the problem of lack of machine resources even for modern powerful computers. In general this problem is a particular case of the problem of minimization of the cost of surface reconstruction. The second problem can be formulated as "What to do, if an algorithm is not capable to reconstruct a CAD-model completely?" This problem usually arises when we deal with clouds of points obtained outside of a laboratory. The problems are found both: separately and together.

A. Used system of denotations

In the given work the following system of denotations is used. The six groups of denoted objects are defined: definitions, conditions, lemmas, mathematical expressions, figures, and tables. Each object is denoted by the index that is composed in the following way. As its basic part the index of corresponding enumerated block of text is used. Before the basic part there is a letter denoting the type of given object: "D" for definitions, "C" for conditions, "L" for lemmas, "E" for expressions, "F" for figures, and "T" for tables. At the end of basic part the ordinal number of the object is added. Enumeration is made from the beginning in each enumerated block of text. For each group of objects enumeration is made separately. If a figure or an expression has concern only to a given definition, condition, or lemma, then the index of such figure or expression is assembled by adding the letter "F" or "E" respectively before the index of given definition, condition, or lemma. For new introduced or locally defined terms the *italic* type of font is used.

In most cases surface reconstruction implies that a given algorithm takes a cloud of sampled points as an input data and produces a CAD-model as a result. This scheme (let's denote it *SI*) is shown in figure 1.

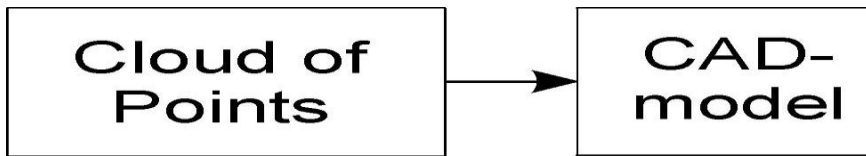


Figure 1. CAD Model

Definition D1.3-1. Let's consider an unsuccessful result of work of some surface reconstruction algorithm. Let we have a particularly reconstructed surface, and all regions of this surface (let's denote the aggregate of them A) are topologically equivalent to the corresponding regions of the original surface. Let's assume there are no invalid edges and triangles in the considered result. It means that each point of the input cloud of points is included in A or doesn't have any triangles and edges (let's call such point a *free point*). Let's call such result an *incomplete normalized CAD-model (ICADM)*. Let's denote the aggregate of regions supplementing A up to a topologically correct CAD-model \underline{A} .

In the given work surface reconstruction is considered as a process of the two following steps: firstly from a given point cloud we obtain an ICADM and then from the ICADM we obtain a CAD-model. This scheme (let's denote it S_2) is shown in figure 2.

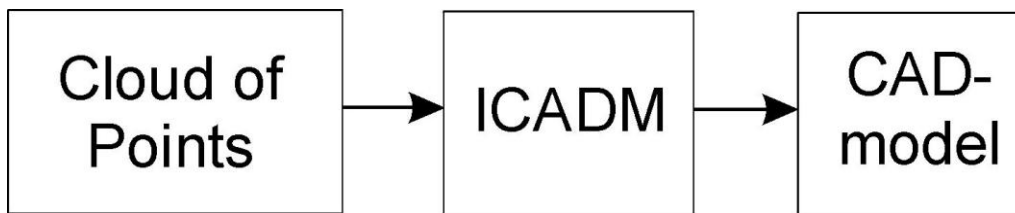


Figure 2. ICADM

The above is the base of scheme S_2 definition of a general surface reconstruction concept is made. Then the case of application of scheme S_1 , when surface reconstruction from a given point cloud can be made by application of a given algorithm with the corresponding cost, is considered. It is shown, that using scheme S_2 , when the given algorithm is used only to realize step ICADM->CAD-model, the cost of the surface reconstruction can be reduced (with the assumption that the given algorithm can process a set of scattered points with given surface boundaries). In the given chapter two simple and fast algorithms to realize step Point Cloud->ICADM are proposed without detailed considering step ICADM->CAD-model. The proposed solution can be used as a pre-processing step for any surface reconstruction algorithm that corresponds to the abovementioned assumption.

The application of scheme S_2 is the only possible way, is considered. It occurs when a used surface reconstruction algorithm can't reconstruct a CAD-model completely. In this chapter a method to realize step ICADM->CAD-model is offered. For surface reconstruction inside damaged regions this method uses force lines of a specially constructed tensor field. The method can be used as a post-processing step for any surface reconstruction algorithm.

II. PROPOSED METHODOLOGY & ASSUMPTIONS

A. Concept of consecutive application of algorithms

Let's consider a way to decrease the total cost of surface reconstruction. Let we have a cloud of N points and two surface reconstruction algorithms (let's denote them A and B respectively). Let algorithm A is able to reconstruct a correct CAD-model with the specific cost c_A (seconds per a point). Let algorithm B can produce a result in that kN ($0 < k < 1$) sampled points belong to a correct triangulated surface and the remaining $(1 - k)N$ sampled points don't belong to it (such points can have edges and triangles or remain free). Let the specific cost of work of algorithm B is c_B (seconds per a point). Let we have also an algorithm of filtering, that having an unsuccessful result of work of a surface reconstruction algorithm at the input can remove all wrong edges and triangles (i.e. produce an ICADM) with the specific cost c_F (seconds per a point). The usually way of processing the given cloud of points is using only algorithm A . In this case the total cost of surface reconstruction is:

$$C_A = c_A N$$

At the same time surface reconstruction can be made in that way: first we apply algorithm B , then we apply the filtering algorithm, and then for the set of remaining free points we use algorithm A (it is supposed that A can process a set of scattered points with given surface boundaries). The cost of such surface reconstruction is:

$$C_{BA} = c_{S2}N + c_{F2}N + (1 - \frac{1}{k})N$$

Definition D2.1-1. Let's call the described above using scheme S2 the *concept of consecutive application of surface reconstruction algorithms (CCAA)*. Let's call an algorithm applied first for processing an input cloud of points the *start algorithm*, an algorithm applied to remove wrong edges and triangles (to obtain an ICADM) the *filtering algorithm*, and an algorithm applied to obtain a CAD-model from the given ICADM the *base algorithm*. This concept can be inductively extended to an arbitrary number of surface reconstruction algorithms having right properties.

Condition C2.1-1. Application of CCAA is more advantageous than application of only the base algorithm if the following condition is fulfilled:

$$kc_A > c_B + c_F \text{ (EC2.1-1)}$$

where all the denotations correspond to the described above example.

For realization of CCAA we need to find for a given algorithm chosen as the base algorithm an algorithm to use as the start algorithm and an algorithm to use as the filtering algorithm with the properties, which meet condition C2.1-1. Since nowadays very many such algorithms are designed, we, of course, in many cases can find suitable existing algorithms. However, let's take into consideration, that the absolute majority of existing surface reconstruction algorithms is designed for self-dependent complete surface reconstruction. But, in CCAA the start algorithm has "a right of mistake". Thus, using this fact we can try to design a surface reconstruction algorithm with the specific cost smaller than existing algorithms. Of course, such algorithm in general case can't be used as a self-dependent surface reconstruction algorithm, but it can be successfully applied as the start algorithm within the framework of CCAA. An algorithm designed in accordance with these means and an algorithm to use as the filtering algorithm are described below.

B. Proposed algorithm: An algorithm to use as the start algorithm

In general, this algorithm uses the ideas of *greedy triangulation (GT)* and belongs to the group of interpolating methods. The GT of a point set in the plane is the triangulation obtained by starting with the empty set and at each step adding the shortest compatible edge between two of the points [1]. In 2D a compatible edge is defined to be an edge that crosses none of the previously added edges. This algorithm is an extension of 2D GT-strategy for 3D.

For using GT-strategy in 3D a special very simple and fast test was designed. This test analyzes a topology of a created mesh at the place of prospective inclusion of an edge, and can be formulated as follows: if insertion of the current tested edge leads to at least one of the following situations:

- appearance of edges having more than two adjacent triangles;
- appearance of a tetrahedron;
- appearance of angles lesser than a given threshold value (let's denote it A_T)

then the edge is considered as incompatible and compatible otherwise. The given test uses floating-point operations in a minimal degree, and is performed fast.

To achieve the linear complexity, the step of sorting created set of all the possible edges is realized in that way: we use a one-dimensional regular grid and during the sorting just put a current considered edge in the corresponding cell of the grid. Such grid having about 1000 cells provide a satisfactory quality of sorting.

Naturally, an algorithm based on such simple principles can't provide 100% reliability. For example, the applied test can't prevent appearance of edges having less than 2 adjacent triangles (edge in figure 3). During experiments we have about 85-98% correctly connected points. But it is enough index for the start algorithm.

C. An algorithm to use as the filtering algorithm

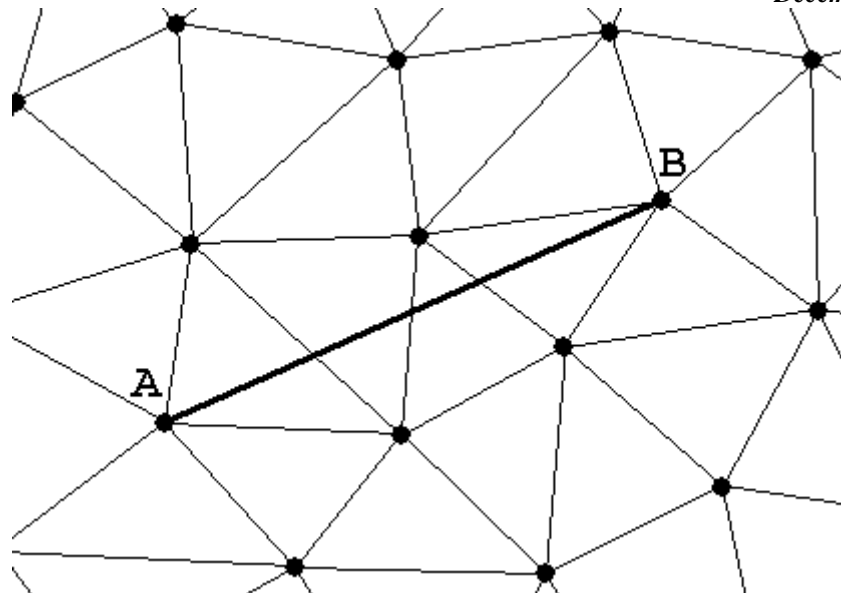


Figure 3.

At the input of the described algorithm we have an unsuccessful result of work of a surface reconstruction algorithm. At the output we have a set of correctly reconstructed regions and a set of free points. In general, the algorithm detects and marks points of correctly triangulated regions. Simultaneously the algorithm removes wrong edges and triangles. The algorithm can improve some errors of the input triangulation, which are caused by generation of redundant edges and triangles, but it doesn't create new edges and triangles.

Condition C2.4-1. On an input set of edges is imposed the following condition: each edge can have only one or two adjacent triangles.

As the basis of this algorithm a variant of the well-known "umbrella" filtering [2] is used. Each point at the input can have one or several chains of adjacent triangles. These chains can be closed or open. Because of condition C3.4-1 such chains can't have any intersections (shared edges). The possible cases are shown in figure 2

Condition C2.4-2. Can be easily proved, that a point of a correctly triangulated region can have only one chain of triangles. This chain is closed for an internal point, and is open for a boundary point.

Condition C2.4-3. Condition C2.4-2 is necessary, but is sufficient only if a given triangulated region is enough.

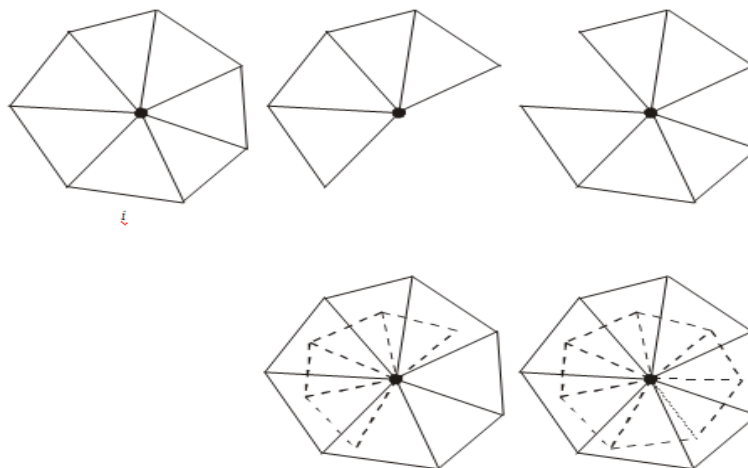


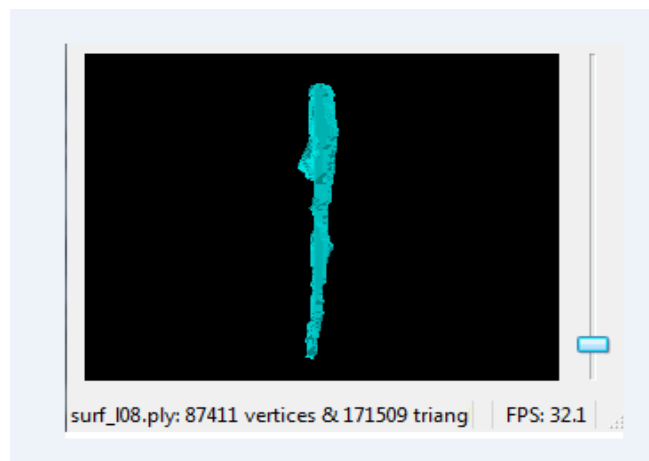
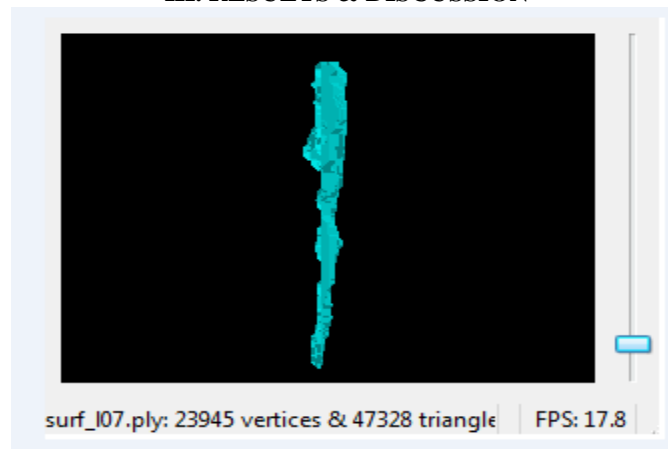
Figure 4.

	Case Action	
1	Only one closed chain	The point is considered an internal point;
2	Only one open chain	If the point meets condition C2.4-3 then this point is considered a boundary point, otherwise all the triangles of the point are deleted and the point is considered a free point;
3	Several open chains	Elimination of all the open chains is made with exception of the chain having the greatest number of triangles, then the point is considered like in case 2;
4	One closed and one or several open chains point is	Elimination of all the open chains is made, then the point is considered an internal point;
5	More than one closed chains	Elimination of all the chains is made, and then the point is considered a free point.

Table 1.

In all the cases of elimination of triangles the described algorithm is recursive applied for all the other points, which were vertices of the deleted triangles.

III. RESULTS & DISCUSSION



The proposed CCAA has been tested in the following configuration, the algorithm described in section:A with AT=11 degree as the start algorithm and the algorithm described in the section :B with Ni=10(introduced in C2.4-3) As the filtering algorithm have been used. All the tests have been done on a computer with 2.0 ghz core i3 CPU.

The used algorithm have the following specific costs:

$$C_b = 1.6 \cdot 10^{-4} S$$

$$C_F = 2.8 \cdot 10^{-5} S$$

$$C_A = 2.7 \cdot 10^{-4} S$$

REFERENCES

1. Hanbo Liu; Xin Wang; Wenyi Qiang, "Implicit surface reconstruction from 3D scattered points based on variational level set method," Systems and Control in Aerospace and Astronautics, 2008. ISSCAA 2008. 2nd International Symposium on , vol., no., pp.1-5, 10-12 Dec. 2008
2. Taku Itoh; Yoshifumi Kanda, "Surface reconstruction from 3D scattered points with normals using both delaunay tetrahedralization and implicit function," System Simulation and Scientific Computing, 2008. ICSC 2008. Asia Simulation Conference - 7th International Conference on , vol., no., pp.780-785, 10-12 Oct. 2008
3. Jun Yang; Changqian Zhu; Hua Zhang, "Surface Reconstruction with Least Square Reproducing Kernel and Partition of Unity," Artificial Reality and Telexistence--Workshops, 2006. ICAT '06. 16th International Conference on , vol., no., pp.375-380, Nov. 29 2006-Dec. 1 2006
4. Xinpeng Ji; Xiaojun Wu; Michael Yu Wang, "Algorithm of Scattered Data Reduction for Surface Reconstruction using Radial Basis Function," Artificial Reality and Telexistence--Workshops, 2006. ICAT '06. 16th International Conference on , vol., no., pp.501-506, Nov. 2006
5. Reuter, P.; Joyot, P.; Trunzler, J.; Boubekeur, T.; Schlick, C., "Surface reconstruction with enriched reproducing kernel particle approximation," Point-Based Graphics, 2005. Eurographics/IEEE VGTC Symposium Proceedings , vol., no., pp. 79-87, 20-21 June 2005
6. Mencl, R.; Muller, H., "Graph-based surface reconstruction using structures in scattered point sets," Computer Graphics International, 1998. Proceedings , vol., no., pp.298-311, 22-26 Jun 1998
7. Garcia, M.A., "Efficient surface reconstruction from scattered points through geometric data fusion," Multisensor Fusion and Integration for Intelligent Systems, 1994. IEEE International Conference on MFI '94. , vol., no., pp.559-566, 2-5 Oct 1994
8. Jalba, A.C.; Roerdink, J.B.T.M., "Efficient Surface Reconstruction From Noisy Data Using Regularized Membrane Potentials," Image Processing, IEEE Transactions on , vol.18, no.5, pp.1119-1134, May 2009
9. Guan, Yudong; Li, Yan; Wang, Ying; Zhang, Hong, "Study of Surface Reconstruction Based on Contours of CT Image Slices," Intelligent Systems and Applications, 2009. ISA 2009. International Workshop on , vol., no., pp.1-4, 23-24 May 2009
10. Sheng, Luo; li-qu, Lin; Xiao-main, Chen, "A New Mesh Growing Surface Reconstruction Algorithm," Education Technology and Computer Science, 2009. ETCS '09. First International Workshop on , vol.3, no., pp.889-893, 7-8 March 2009
11. Zhan Zongqian; Zhang Xiaoqian, "3D surface reconstruction and panorama stitching based on LCD-based calibration and multi-baselines stereo matching," Image Analysis and Signal Processing, 2009. IASP 2009. International Conference on , vol., no., pp.278-281, 11-12 April 2009
12. Andriy, P.; Bohdan, R., "Application of methods of 3D surface reconstruction for characterization of pitting defects," CAD Systems in Microelectronics, 2009. CADSM 2009. 10th International Conference - The Experience of Designing and Application of , vol., no., pp.468-468, 24-28 Feb. 2009
13. Lasser, T.; Soubret, A.; Ripoll, J.; Ntzachristos, V., "Surface Reconstruction for Free-Space 360 \circ Fluorescence Molecular Tomography and the Effects of Animal Motion," Medical Imaging, IEEE Transactions on , vol.27, no.2, pp.188-194, Feb. 2008
14. de Medeiros Brito, A.; Doria Neto, A.D.; Dantas de Melo, J.; Garcia Goncalves, L.M., "An Adaptive Learning Approach for 3-D Surface Reconstruction From Point Clouds," Neural Networks, IEEE Transactions on , vol.19, no.6, pp.1130-1140, June 2008
15. Lianwei Zhang; Tingbo Hu; Xiaolin Liu; Yan Li; Tao Wu; Hangen He, "Surface reconstruction from cloud points based on Support Vector Machine," Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on , vol., no., pp.377-381, 1-3 Sept. 2008
16. Bin Forkan, F.; Shamsuddin, S.M.H., "Kohonen-Swarm Algorithm for Unstructured Data in Surface Reconstruction," Computer Graphics, Imaging and Visualisation, 2008. CGIV '08. Fifth International Conference on , vol., no., pp.5-11, 26-28 Aug. 2008
17. Xiaoqiang Wang; Zhuangde Jiang; Bing Li, "Surface reconstruction based on quasi-ordered point cloud," Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on , vol., no., pp.306-310, 1-3 Sept. 2008
18. Bilir, S.C.; Yemez, Y., "Time varying surface reconstruction from multiview video," Shape Modeling and Applications, 2008. SMI 2008. IEEE International Conference on , vol., no., pp.47-51, 4-6 June 2008

19. Trehan, A.; Amineh, R.K.; Georgiev, M.S.; Nikolova, N.K., "Accuracy assessment of photogrammetry surface reconstruction for improving microwave imaging," Antennas and Propagation Society International Symposium, 2008. AP-S 2008. IEEE , vol., no., pp.1-4, 5-11 July 2008
20. Bilir, S.C.; Yemez, Y., "Time varying surface reconstruction from multiview video," Signal Processing, Communication and Applications Conference, 2008. SIU 2008. IEEE 16th , vol., no., pp.1-4, 20-22 April 2008
21. Wu Fuzhong, "Surface Reconstruction Method Based on GRNN," Intelligent Computation Technology and Automation (ICICTA), 2008 International Conference on , vol.1, no., pp.262-265, 20-22 Oct. 2008
22. Xinrong Hu; Naixue Xiong; Yang, L.T.; Dehua Li, "A Surface Reconstruction Approach with a Rotated Camera," Computer Science and its Applications, 2008. CSA '08. International Symposium on , vol., no., pp.72-77, 13-15 Oct. 2008
23. Taku Itoh; Yoshifumi Kanda, "Surface reconstruction from 3D scattered points with normals using both delaunay tetrahedralization and implicit function," System Simulation and Scientific Computing, 2008. ICSC 2008. Asia Simulation Conference - 7th International Conference on , vol., no., pp.780-785, 10-12 Oct. 2008
24. Sun, P.P.; Jin Jin Zheng; Zhou, H.J.; Shen, L.G., "3D B-Spline surface reconstruction based on the contour information," Information and Automation, 2008. ICIA 2008. International Conference on , vol., no., pp.1697-1702, 20-23 June 2008
25. Goinski, A., "Evolutionary surface reconstruction," Human System Interactions, 2008 Conference on , vol., no., pp.464-469, 25-27 May 2008
26. Fuhua Cheng; Fengtao Fan; Conglin Huang; Jiayi Wang; Shuhua Lai; Miura, K.T., "Chapter 5: Smooth Surface Reconstruction Using Doo-Sabin Subdivision Surfaces," Geometric Modeling and Imaging, 2008. GMAI 2008. 3rd International Conference on , vol., no., pp.27-33, 9-11 July 2008
27. Gang Hu; Hao Li; Junjie Yao; Jing Bai, "Fluorescent Optical Imaging of Small Animals Using Filtered Back-projection 3D Surface Reconstruction Method," BioMedical Engineering and Informatics, 2008. BMEI 2008. International Conference on , vol.2, no., pp.76-80, 27-30 May 2008
28. Krumnikl, M.; Sojka, E.; Gaura, J.; Motyka, O., "A New Method for Bryophyte Canopy Analysis Based on 3D Surface Reconstruction," Computer Information Systems and Industrial Management Applications, 2008. CISIM '08. 7th , vol., no., pp.210-211, 26-28 June 2008
29. Harker, M.; O'Leary, P., "Least squares surface reconstruction from measured gradient fields," Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on , vol., no., pp.1-7, 23-28 June 2008
30. Xue-mei Wu; Gui-xian Li; Wei-min Zhao, "Incomplete Points Cloud Data Surface Reconstruction Based on Neural Network," Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP '08 International Conference on , vol., no., pp.913-916, 15-17 Aug. 2008