Measures for Surface Comparison on Unstructured Grids with Different Density

Natalia Dyshkant

Lomonosov Moscow State University,
Faculty of Computational Mathematics and Cybernetics,
Leninskie gory, CMC MSU, 119991 Moscow, Russian Federation
natalia.dyshkant@gmail.com

Abstract. We consider the problem of surface comparison given as spatial point clouds that can be explicitly projected onto a plane. This problem can be reduced to comparison of mesh functions of two variables given on different grids. A general case when both grids are unstructured and have different density is of interest. A measure to compare such functions that allow to estimate difference on areas with nodes from both grids and an algorithm to compute it are proposed. Estimation for computational complexity of the algorithm is presented. Computing experiments on real data (3d face models) were carried out.

Keywords: discrete surface model, metrics for surface comparison, unstructured grid, Delaunay triangulation, minimum spanning tree, 3d face image.

1   Introduction

With the rapid progress of modern 3d scanning technologies [1], objects' surfaces can be routinely acquired as discrete surface models, which consist of clouds of points with 3d coordinates. Spatial objects' shape can be considered as a set of *schlicht* surfaces, i.e., such that can be uniquely projected onto a plane (see Fig. 1). In this paper, new measures for comparison of such surfaces and effective methods to compute them are devised.

Exact and computationally efficient algorithms for computing disparity measures between surfaces are required in many applications of image analysis and computer graphics. Surface comparison and matching methods are needed to solve problems of surface classification, registration, reconstruction by its separate fragments, etc.

Raw schlicht surface data acquired by 3d scanner can be considered as a discrete function defined at nodes of (generally speaking) *unstructured* grid. This means that nodes of a grid can be spread randomly, i.e., there is no ordered structure in the grid.

There are two main presentation methods for modelling of schlicht surfaces: definition of surface on structured (uniform) and unstructured grids (see Fig. 1).

The first method is tedious in case of requirement for sufficiently detailed presentation of surface. Besides, a selection problem of cell size of a grid appears. The size should be optimal to get adequate accuracy of surface approximation for concrete applied problem. In case of unstructured grids, surface approximation quality is higher. At the same time it is required to introduce and design more complex measures for surface comparison given on *different* grids and processing algorithms.
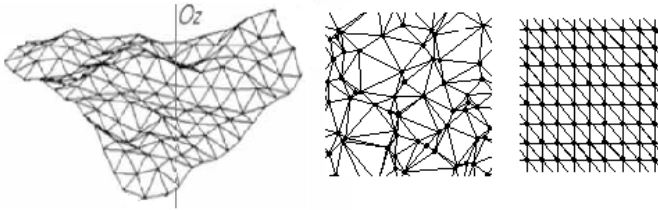

Fig. 1. Examples of schlicht surface (on the left), unstructured grid (in the middle) and uniform (structured) grid (on the right)

At the first stage of the most proposed approaches for surface comparison two source unstructured grids are transformed to a general uniform grid [2]. Then a measure between two surfaces is computed. At the transformation stage interpolation and extrapolation of source data are used. Therefore surface approximation quality decreases. A lot of methods are too computationally intensive for achievement of adequate accuracy. In this paper, we propose the approach, which saves given nonregularity of grids.

In [3] the author proposed a measure for comparison of functions, which will be discussed further. The proposed algorithm to compute the measure used Delaunay triangulations of each point clouds, general Delaunay triangulation constructed for both clouds, function interpolation on basis of localization of triangulations in each other and function comparison on single cells of general triangulation. Localization was implemented on basis of minimum spanning trees. In this paper, we concentrate on modifications of this measure for case when source grids are not only unstructured but also have different density and designing an effective algorithm to compute new measures.

In practise, such measures are needed for comparison of surfaces acquired by 3d scanners of different accuracy: there are applications for 3d face model matching or matching of 3d face model of person and his 3d jaw models in orthodontics [4]. Jaw models are acquired by special orthodontic scanner of very high accuracy, 3d scanner for acquisition of face models is of less accuracy.

This paper is organized as follows. A review of surface comparison problem, formalized problem statement and the proposed measures are given in section 2. The proposed algorithm for measure computing and evaluation of its complexity are given in section 3. Section 4 presents experimental results for comparison of 3d face surfaces. Conclusion is given in section 5.

2   Surface Comparison Problem

Though there has been a great deal of progress in 3d object comparison field, computationally efficient and accurate surface comparison is still an urgent and unsolved problem.

2.1   Related Work

A large body of research addresses not to discrete surface model but to initial continuous surface representation or definition of surface on a uniform grid [5], [6]. In [5] measure for comparison of surfaces f_1, f_2 defined on grids g_1, g_2 was introduced as the following:

$$\rho(f_1, f_2) = \max_{g_1[i] \in g_1} \min_{g_2[j] \in g_2} d(g_1[i], g_2[j]),$$

Some researches proposed methods of referencing or comparison by descriptors. For example, in [7] the objects' comparison problem was reduced to graph comparison problem. Vertices of graph represent separate fragments of object's surface and edges represent information about their connectivity.

In [8] the problem of surfaces given on different point sets was considered. Distance from a point of one surface to the other one was computed along a normal to the nearest spline for this point. Comparison measure based on computing of distance difference along surface normals is interesting as it doesn't require transformation of source grids to the common one.

2.2   Mathematical Problem Statement

A 3d schlicht surface S given as a point cloud $\{(x_i, y_i, z_i)\}_{i=1}^{N}$ can be considered as the one-valued function $z = F(x, y)$ defined on the discrete set $\{(x_i, y_i)\}_{i=1}^{N}$. We construct plane Delaunay triangulation of this set. *Triangulated surface model* consists of spatial triangles defined by the values z_i at nodes (x_i, y_i) of this triangulation. Such piecewise linear model interpolates the initial surface S.

Definition 1. *A finite point set* $G : \{(x^i, y^i) \in \mathbb{R}^2 | i = 1, \ldots, N\}$, $N \geq 3$ *is called a nonuniform two-dimensional grid.*

Suppose R is a rectangle in $\mathbb{R}^2$, $\mathcal{G}$ is the set of nonuniform two-dimensional grids contained in R, $\mathcal{F}$ is the set of single-valued functions given on grids from $\mathcal{G}$.

Consider the following problem statement.

Let S_1, S_2 be two schlicht surfaces defined by discrete functions $F_1, F_2 \in \mathcal{F}$ at the nodes of grids $G_1, G_2 \in \mathcal{G}$, respectively, $N_1 = |G_1|, N_2 = |G_2|$. It is required to introduce measures for comparison of surfaces S_1, S_2 and design an approach to compute them.

Note that two surfaces are given on two *different* grids and it is not possible to compute a measure between them directly. So the basic idea of the proposed approach is to fill "missing" values of each surface at the nodes of the other grid.

Let $\hat{F}$ be a continuous piecewise linear function in $\mathbb{R}^2$. We say that $\hat{F}$ *approximates* F at G if $\hat{F}(x,y) \equiv F(x,y)$ for all $(x,y) \in G$. By $\hat{\mathcal{F}}$ denote the set of continuous functions in $\mathbb{R}^2$ that approximates functions from $\mathcal{F}$.

Denote by T_1 and T_2 the Delaunay triangulations constructed on grids G_1 and G_2, respectively. The Delaunay triangulation constructed on the union of two grids $G = G_1 \cup G_2$ is called the *general* (or *united*) Delaunay triangulation and is denoted by T.

2.3   Proposed Measures

Let ρ be a function in $\mathcal{F}$ such that the following conditions hold: $\forall F_1, F_2, F_3 \in \mathcal{F} : \rho(F_1, F_2) \geq 0,\ \rho(F_1, F_2 + F_3) \leq \rho(F_1, F_2) + \rho(F_1, F_3)$. We shall say that ρ is a difference functional of two functions (schlicht surfaces).

Consider a function $\mu(x, y)$ that defined *weight* of difference between two surfaces at a certain point with coordinates (x, y) in accordance with significance of function similarity in the region contained this point. Let A, B, C be nodes of the united grid G. By definition, put:

$$V_\mu(A, B, C, F_1, F_2) = \iint\limits_{\triangle ABC} \left| \hat{F}_1(x,y) - \hat{F}_2(x,y) \right| \mu(x,y)\, dxdy. \qquad (1)$$

In [3], the author proposed the following measure:

$$\rho_{V_\mu}(F_1, F_2) = \sum_{\triangle ABC \in T} V_\mu(A, B, C, F_1, F_2)/S_{\triangle ABC}, \qquad (2)$$

where weighted difference volume is summarised over all triangles $\triangle ABC$ of the general triangulation and each summand is normalised by the area of triangle $S_{\triangle ABC}$.

By introducing of function $\mu(x, y)$ on surfaces the defined measure can be adapted for concrete applied problems. For example, in case of face models' comparison, μ can have greater value on regions where skin is close with a skull than on another regions with more tissues.

In fact, measure (1)-(2) is an analog of L_1-metric for interpolated source functions $\hat{F}_1$ and $\hat{F}_2$.

In case $\mu(x, y)$ is the identical function, similarity of all surface patches allows with equal weight. But objective nature of data is that source grids are not always uniform. There often exists such regions, which have sufficiently great area and consist of nodes from one of the grids only.

Let V be a value of V_μ for $\mu(x, y) \equiv 1$; suppose $\rho_V(F_1, F_2)$ is measure (2) for condition of $V_\mu = V$: $V = V_\mu\big|_{\mu \equiv 1}$, $\rho_V = \rho_{V_\mu}\big|_{V_\mu = V}$.

The defect of ρ_V is that it allows both type of regions where nodes from two grids are mixed with one another and where one of the grids is more thick than the other using the same weight. Such measure can be successfully used in case when nodes of both grids are distributed uniformly with roughly equal density.

In the considered case, when grids are unstructured and have different density this defect is revealed. To get out from it we propose a measure that takes in account only representative data, i.e., regions, where nodes of both grids are concentrated:

$$\rho_{\delta V}(f_1, f_2) = \sum_{\substack{\triangle ABC \in T: \\ \triangle ABC \notin T_1, \\ \triangle ABC \notin T_2}} V(A, B, C, f_1, f_2)/S_{\triangle ABC}. \tag{3}$$

We claim that $\rho_{\delta V}(f_1, f_2) \equiv \rho_{V_\mu}(f_1, f_2)$ for

$$\mu(x, y) = \begin{cases} 1, & (x,y) \in \triangle ABC: \ \triangle ABC \in T, \triangle ABC \notin T_1 \text{ or} \\ & \triangle ABC \notin T_2; \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Indeed, this follows from definition of ρ_{V_μ}.

3   Methods

During merging process of two Delaunay triangulations T_1 and T_2, some edges and triangles move to the united triangulation without changes and some of them are destroyed. So there are new edges and triangles, which connect nodes from different grids, in the general triangulation T.

We say that an edge or a triangle is called *interface* if it connects nodes from both of grids G_1, G_2. On Fig. 2 all interface triangles constructed during merging are filled.

Measure (3) is calculated over interface triangles only. Naive algorithm to compute this measure examines each triangle and calculates difference volume (1) if this triangle is interface. For effective calculation of the measure more complex algorithm for extracting of interface triangles is needed.

Note also that in case when nodes of two grids are well mixed with one another all triangles of T can be interface. In this case values of measures (2) and (3) are equal. The advantage of measure (3) appears in cases when one surface is presented in more detail than the other. For example, in practise one can meet this case solving surface matching problem for surfaces acquired by two 3d scanners of different accuracy.
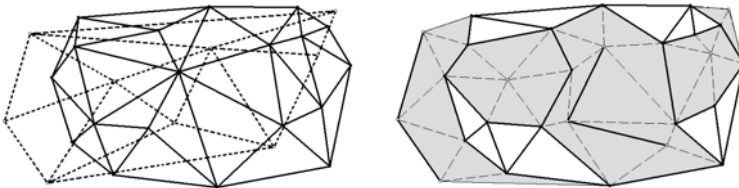

Fig. 2. Two triangulated grids (on the left) and united Delaunay triangulation with fill interface triangles (on he right)
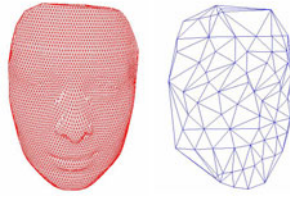
Fig. 3. Example of face models with different levels of detail

3.1   Algorithm for Interface Triangles Extraction

A set of interface triangles decomposes on several subsets. Each subset is a chain of triangles, which are pairwise incident by edges. This chain can be closed or open-ended (see Fig. 4). In the second case each of two terminal triangles has at least one boundary side, i.e., side from a convex hull $Conv(G)$ of G.
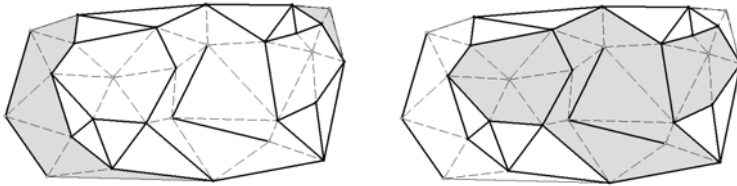

Fig. 4. Two open-ended chains (on the left) and three closed chains of interface triangles (on the right)

Therefore an algorithm for interface triangles extraction reduced to tracing of chains mentioned above.

Let us introduce concepts of connection for triangulation elements.

Definition 2. *A set of nodes and edges (edges and triangles) of triangulation is called connected if for any pair of its nodes (edges) there exists a chain consisted of pairwise incident nodes and edges (edges and triangles).*

Definition 3. *A maximal connected set of nodes and edges is said to be a clout if it moves completely (without changes) to general triangulation T from one of the source triangulations.*

Let us remark that the problem of chain tracing can be considered as the problem of *clout location in triangulation*.

We say that an interface edge of general triangulation T is called a *starter* if it belongs to a chain that is not traced yet. In other words, a starter initializes the process of chain tracing.

Hence the proposed algorithm for interface triangles extraction consists of the following stages:

1. Search for initial starter;
2. Tracing of chain conformable to the found starter;
3. Search for the next starter. If it exists move to the previous stage, otherwise finish.

Let us consider each stage in detail.

Search for First Starter. Assume that nodes of G_1 and G_2 are lexicographically ordered and $g_1[0]$, $g_2[0]$ are the leftmost nodes of grids. Without loss of generality, we can assume that $g_1[0] \prec g_2[0]$.

Let us consider a circle that passes through $g_1[0]$ and $g_2[0]$ of center on horizontal ray with origin at $g_2[0]$ directed to the left.

Now we shall give the following definitions. A circle is called *empty* with respect to the grid G if it doesn't contain nodes of G. A circle is *incident to a point* if it passes through this point. A circle incident to a point is called a *maximal empty circle* if it is empty and has maximal radius. We say that an edge satisfies the *Delaunay condition* if there exists an empty circle that passes through the endpoints of this edge.

By construction, the considered circle is empty w.r.t. G_2. At the same time it can contain some nodes $g_1[i_1], \ldots, g_1[i_n]$ from G_1. Let us construct a set of n different circles of center at the same horizontal ray that passes through one of these nodes. Choose a node $g_1[k] \in g_1[i_1], \ldots, g_1[i_n]$ such that it conforms to the circle of minimal radius from the constructed set (see Fig. 5).
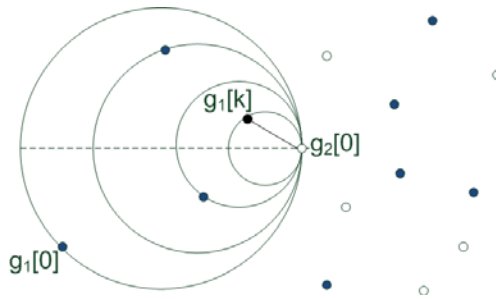

Fig. 5. Search for initial starter

The circle of minimal radius is empty w.r.t. G_1 and G_2, consequently w.r.t. G. It follows that the edge connected nodes $g_1[k]$ and $g_2[0]$ satisfies the Delaunay condition and belongs to general triangulation T. The edge $g_1[k]g_2[0]$ is the *first starter* because the conditions of the following theorem hold.

Theorem 1. *(starter existence criterion) A node $g_1[i]$ of a grid G_1 has an incident interface edge of the general Delaunay triangulation constructed on $G_1 \cup G_2$ iff there exists a circle C such that the following conditions hold*

(i) C is incident to this node;

(ii) *C is empty w.r.t. the set $G_1 \setminus \{g_1[i]\}$;*

(iii) *C contains inside or on the boundary at least one node of the second grid G_2.*

Proof. Necessity. If two nodes of different grids G_1, G_2 form the interface edge of T then they have the incident empty circle that satisfies all required conditions by definition.

Sufficiency. Suppose $v \in G_1$ and there exists the circle such that it is incident to v, contains inside and/or on its boundary nodes $u_1, \ldots, u_m \in G_2$ but is empty w.r.t. other nodes of G_1. Let us consider a set of m embedded circles that pass through pairs of nodes v, u_i, $i = 1, \ldots, m$ and have the common with the initial circle tangent at v. The circle from this set of minimal radius is incident to $v \in G_1$, $u_k \in G_2$ and empty w.r.t. $G_1 \cup G_2$. Therefore nodes v and u_k form the interface edge, which can be chosen as a starter.

Search time for the first starter is composed of time to compute the leftmost nodes of both grids and single look-up of nodes from G_2 during circle of minimal radius search. Thus time is linear in number of nodes in united grid.

Tracing of Chain of Interface Triangles. During tracing process sequential extraction of interface edges occurs. An interface edge extracted on a certain iteration is declared as *current*. The first starter becomes the first current edge.

Let AB be a current edge. During joining of edges AB and BA to lists of incident nodes for A and B, respectively, edges of these lists (bundles) that don't satisfy the Delaunay condition must be destroyed. Hence *bundle correction procedure* for interface edge is needed (see Fig. 6). It is based on checking the Delaunay condition by angle criterion [9].

The tracing process for chain using the found starter consists of the following steps:

a) To declare the starter as a *current* edge;

b) To correct bundles of endpoints of the current edge in the triangulations T_1, T_2 (destroy edges that don't satisfy the Delaunay condition);

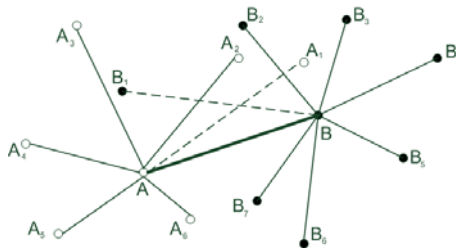c) To construct a *new* edge that connects nodes of different grids;

Fig. 6. Bundle correction for an interface edge AB. Edges AA_1 and BB_1 are destroyed

d) To destroy edges in T_1 and T_2 that intersect the triangle formed by he current and new edges;
e) To declare the new edge as a *current* and go to (b).

The tracing algorithm has analogy with an algorithm for Delaunay triangulation construction, which consists of cleaning and sewing together stages, from [10].

If on step (c) a new edge coincides with the starter then the traceable chain is closed. If we don't succeed in new edge construction then the current edge is a terminal edge of the chain. In the last case it is needed to continue the tracing process from the starter to the other side until the second terminal edge is found.

Search for Next Starter. Search of the rest of the starters is implemented using minimal spanning trees (MST) for Delaunay triangulations T_1, T_2 as if during chain tracing process connection of Delaunay triangulation is disturbed (see Definition 2) then there must be an edge of MST in the set of destroyed edges.

Let us introduce some definitions. We say that a *bridge* is an edge of MST for T_1 or T_2 such that it was destroyed during chain tracing process. An *influence circle of a bridge* is a circle such that this bridge is a diameter of it. One of the endpoints (nodes) of a bridge belongs to already traced chain and the other one is still a *free node*.

Lemma 1. *For any free node there exists an incident starter.*

In these terms, a free node, which has an incident starter for initialising chain tracing process, is generated during bridge destroying.

The following lemma is used to look for the second node of the starter:

Lemma 2. *Suppose B is a free node, A is a node of a triangulation T, a pair A, B forms a starter. Then B is inside of at least one of the maximal empty circles in T that are incident to A.*

Search for starter using the free node A and the bridge AB consists of the following steps:

a) Search for traced interface triangle such that there is the free node A in the circumcircle of it (using search along AB);
b) Search for interface triangles such that they are adjacent to the found triangle and there is A inside their circumcircles;
c) To construct a set $D = \{D_1, \ldots, D_n\}$ of nodes of the found triangles that are inside of the influence circle for the bridge AB;
d) To construct a circle of center on AB that passes through the nodes D_i, A for any $D_i \in D$ and select the node D_{i^*} that conforms to the circle of minimal radius (similarly to search for the first starter). The edge $D_{i^*}A$, which connects the selected node with the free one, is a new starter.

3.2   Computational Complexity of the Algorithm

In this subsection we present some auxiliary lemmas for evaluation of complexity of the proposed algorithm.

By the above a bridge is an edge of MST. MST of a triangulation is a subgraph of this triangulation [12]. This means that the influence circle of a bridge is empty w.r.t. endpoints and middle points of this bridge. Using this property we can estimate intersection measure between a bridge and the influence circle of another bridge.

Lemma 3. *A bridge can not cut from the influence circle of another bridge an arc of size more than $60°$.*

Lemma 4. *Suppose AA_1 and BB_1 are bridges of a triangulation T_1, both AA_1 and BB_1 intersect an edge PQ of a triangulation T_2, node P is inside of both influence circles of the bridges AA_1, BB_1. Then different of angles $\angle APA_1$ and $\angle BPB_1$ is not less than $60°$.*

Lemma 5. *Suppose an edge of T_1 intersects several bridges of T_2. Then an endpoint of this edge is inside of influence circles of at the most two bridges.*

This implies that we can estimate number of bridges in T_1 that can be destroyed by an edge of T_2.

Lemma 6. *During merging procedure of two Delaunay triangulations each edge intersects at the most four bridges that it destroys.*

The above four lemmas were proved by Mestetskiy, Tsarik in [13]. These lemmas are needed for proof of the following theorem.

Theorem 2. *Computational complexity of algorithm for interface triangles extraction is $O(N)$, where N is a number of nodes in general grid.*

The proof is omitted.

Let us consider main stages of general algorithm for computing measure (3) (see [14]) and complexity of each stage:

1) Construction of Delaunay triangulations T_1, T_2: $O(N_1 \log N_1) + O(N_2 \log N_2)$;
2) Construction of minimal spanning trees of triangulations using Cherion, Tarjan algorithm (see [15]): $O(N_1) + O(N_2)$;
3) Localization of each of two grids G_1, G_2 in the triangulation of the other grid: $O(N)$ *experimentally*;
4) Interpolation each of two functions F_1, F_2 on the grid that the other function is defined on: $O(N)$;
5) Extraction of all interface triangles: $O(N)$;
6) Summing V over all interface triangles, we obtain value of measure: $O(N)$.

Note that stage (5) doesn't use results of stages (3) and (4). It can be proved that stage (3) can be implemented during or after extraction procedure using its intermediate results. Hence theoretical complexity (not only experimental) of stage (3) is linear.

Therefore total computational complexity for measure computing of surfaces given by discrete models is $O(N \log N)$ because of triangulation construction stage. In case of surfaces are given by triangular models the complexity is $O(N)$.

4   Experiments

Computational experiments were carried out on 3d face models acquired by 3d scanner Broadway^{TM} designed by Artec Group Company [1]. The database consists of 48 models received by scanning of 8 different persons (6 different models for each person). Different models of one and the same person were used as surfaces for comparison.

Suppose S_1, S_2 are surfaces for comparison, S_2' is a reduced (simplified) second surface. S_2' is acquired from S_2 by uniform random thinning of the second grid. As the result of thinning 15% nodes of the second grid were removed.

Grids of S_1, S_2 are unstructured (nonuniform) but has approximately equal density. Hence we assume a value of measure (2) between them as adequate initial estimation. Grids of S_1 and S_2' are unstructured grids with different density.

Table 1 shows an example of measure values for comparison of surfaces from the database. We see that the measure (3) estimates difference between surfaces S_1 and S_2' more adequate than the measure (2). Values of both measures (2) and (3) are bigger than the initial estimation but the difference between (3) and the initial estimation is less. Similar results were received for the rest models of the database.

Table 1. Value of measures (2) & (3) for surface comparison

Measure	Value (mm)	Comments
$\rho_V(S_1, S_2)$	7094	Adequate estimation for S_1 & S_2
$\rho_V(S_1, S_2')$	17963	$\rho_V(S_1, S_2') > \rho_V(S_1, S_2)$
$\rho_{\delta V}(S_1, S_2')$	10884	$\rho_{\delta V}(S_1, S_2') \approx \rho_V(S_1, S_2')$

5   Conclusions

New measure adapted for comparison problem of surfaces defined on unstructured grids with different density is introduced. An efficient algorithm for measure computing is proposed.

The measure allows only surface fragments that are represented by nodes of both grids. We call such fragments interface fragments. For efficiency of measure computing a new algorithm for interface triangles extraction is proposed. Computational complexity of the algorithm is presented.

Computing experiments for comparison of surfaces defined on grids with different density were carried out. As experimental estimations have shown, the introduced measure is adequate for such kind of source grids.

References

1. Artec Group — 3D Scanning Technologies, `http://www.artec-group.com`
2. Kolesov, A., Pavlova, O.: Surfer Package — Processing and Visualization of 2d Functions (in Russian). Computer Press (1999)
3. Dyshkant, N.: Disparity Measure Construction for Comparison of 3D Objects' Surfaces. In: Proceedings of the 2nd International Workshop on Image Mining, Theory and Applications (IMTA-2-2009), pp. 43–52. INSTICC Press, Lisbon (2009)
4. Dzaraev, C., Persin, L., Porochin, A.: Using of 3d Scanners for Diagnostics of Dentoalveolar Anomalies (in Russian). In: Theoretical and Practical Forum "Dental-Review", Moscow (2010)
5. Gruen, A., Akca, D.: Squares 3D Surface and Curve Matching. ISPRS Journal of Photogrammetry and Remote Sensing 59, 151–174 (2005)
6. Cignoni, P., Roccini, C., Scopigno, R.: Metro: measuring error on simplified surfaces. In: Computer Graphics Forum, vol. 17, pp. 167–174. Blackwell Publishers, Malden (1998)
7. Fan, T., Medioni, G., Nevatia, R.: Recognizing 3D objects using surface descriptions. IEEE PAMI 11(11), 1140–1157 (1989)
8. Schenk, T.: Digital Photogrammetry. Terra-Science, Laurelville (1999)
9. Skvortsov, A.: A survey of algorithms for constructing a Delaunay triangulation (in Russian). In: Numerical Methods and Programming, vol. 3. RCC MSU Press (2002)
10. Mestetskiy, L.: Continuous Morphology of Binary Images: Figures, Skeletons and Circulars (in Russian). PHYSMATLIT Press, Moscow (2009)
11. Mestetskiy, L., Tsarik, E.: Delaunay triangulation: recursion without space division of vertices (in Russian). In: 14th International Conference on Computer Graphics and Vision GraphiCon 2004, pp. 267–270. CMC MSU Press, Moscow (2004)
12. Preparata, F., Shamos, M.: Computational Geometry: An Introduction. Springer, Heidelberg (1985)
13. Mestetskiy, L., Tsarik, E.: Merging od Unseparated Delaunay Triangulations. In: Complex Systems: Processing, Modelling and Optimization Od Information, vol. 2, pp. 216–231. Tver State University, Tver (2004)
14. Dyshkant, N., Mestetskiy, L.: Comparison of One-Sheet Surfaces Acquired by 3D Scanning (in Russian). In: 18th International Conference on Computer Graphics and Vision GraphiCon 2008, pp. 270–277. CMC MSU Press, Moscow (2008)
15. Cheriton, D., Tarjan, R.: Finding minimum spanning trees. SIAM J. Comput. 5(4), 724–742 (1976)