

Comparison of Point Clouds Acquired by 3D Scanner

Natalia Dyshkant*

Lomonosov Moscow State University,
Faculty of Computational Mathematics and Cybernetics,
GSP-1, 1-52, Leninskiye Gory, MSU, Moscow, 119991, Russian Federation
natalia.dyshkant@gmail.com

Abstract. 3D representation of real objects surfaces can be used in applications of computer graphics, medicine, geoinformatics, etc. We consider a problem of measure introducing for comparing of point clouds acquired by different scanning acts and types of scanners and designing of computationally efficient algorithms for their computing. The solution supposes estimation of disparity measure for the fixed position and search of such position that the measure is minimal by solving optimization problem of surface matching. The algorithm for efficient localization of mesh nodes in a Delaunay triangulation is proposed. As the applications several problems of 3D face model analysis were considered.

Keywords: Discrete surface model, Delaunay triangulation, Euclidean minimum spanning tree, computational geometry, 3D face image.

1 Introduction

Information obtained from a 3D scanner after scanning of an objects's surface is usually presented as a discrete set of nodes with three-dimensional coordinates and represents a *discrete model* of a surface. The concepts of connectivity, topology, or continuous surface are not specified explicitly for such set. On basis of this set one can receive triangulated polygonal surface model [1], which will be a *continuous model* of a surface.

A literature review of existing techniques for single-valued surface modelling showed that there are two basic approaches to represent such surfaces: definition at the nodes of the regular or irregular meshes. A two-dimensional (plane) *mesh* is a set of mutually connected geometric elements (nodes, edges, and cells). The mesh nodes represent a finite set of points on the plane. Both approaches have their advantages and disadvantages. Method using an irregular mesh allows to adapt to the required accuracy of the surface description and does not contribute redundancy, which increases computing resources, to the source data.

Often during comparison of two surfaces it is assumed that for each node of one surface point there is the corresponding node of the second surface. This

* The reported study was partially supported by RFBR, research projects Nos. 12-07-13307 mol_a, 11-01-00783 a.

assumption implies definition of surfaces at the same discrete set of nodes. Initial surface objects acquired by three-dimensional scanning, have irregular structure. During the transforming stage from the original irregular data to regular mesh one face the challenge of choosing the optimal size for size of mesh cell, which leads to inefficiency of the approach.

The majority of the existing measures for comparing surfaces can be calculated directly if both surfaces are defined at the nodes of some common mesh. However, such measures do not allow extension to the case of surfaces defined on different meshes without the stage of transition to regular meshes. There are measures that can easily be generalized to the case of two different meshes, but their calculation in this case has a quadratic complexity.

The existing methods for mapping the surfaces can be divided into two classes:

- a. Surface adjustment based on calculation of distances between points in the three-dimensional space, which has a large computer complexity.
- b. Recalculation of the original data in a regular two-dimensional mesh, which leads to description redundancy and also to significant increase in computational complexity.

It follows that the problem of developing of new computationally efficient algorithms for comparison of surfaces represented as point clouds, preserving the original irregular meshes remains actual at present.

The paper is organized as follows. A literature review of the problem for comparison of point clouds is given in section 2. The proposed approach for surface comparison is given in section 3. The proposed method of mesh localization in Delaunay triangulation presented in Section 4. Section 5 presents some applied problems for 3D face image analysis considered by the author. Conclusion is given in section 6.

2 Related Work

As it was stated above, many existing approaches for point cloud comparison use the transformation of initial irregular meshes to the common regular mesh [2], [3], [4] (see Fig. 1). After such transformation the approaches of surface comparison and matching for regular sets of nodes [5] can be applied.

During the transformation we face the problem of choosing the optimal step for a regular mesh, which leads to a significant amount of calculations to achieve an acceptable accuracy of the approximation surface and inefficiency. In this paper we propose a method for comparing surfaces with preservation of the original irregularity of meshes.

The most of existing methods is assumed that for each single node of the first surface there is the corresponding node on the second surface [6], [7]. There are methods based on a comparison of feature descriptions [8].

One of the basic algorithms for surface matching is the algorithm of iterative closest points — ICP, proposed in [6], [9], [7]. The algorithm uses an iterative procedure to minimize the average distance between two point clouds. This requires an initial rough estimate of converting one cloud to another, which is

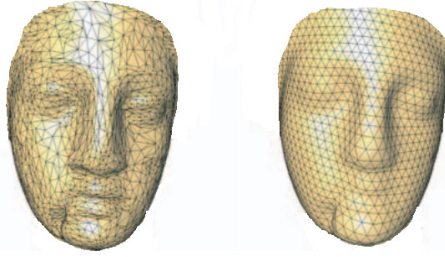


Fig. 1. Transformation of the irregular mesh to the regular one (adoption of illustration from [4])

gradually refined in the process of minimization. For two given point clouds S_1 and S_2 the algorithm finds the transformation from S_1 into S_2 .

The ICP algorithm can be used to align the images of the same object obtained from different angles, which has common areas — regions of overlap. This assumes that there are pairs of closest points such that distances between them are less than a threshold in the regions of overlap. If some "wrong" pairs which do not belong to the region of overlap are included in a list of pairs of closest point, it will negatively affect to the results of the algorithm.

Let $\{(s_i^1, s_i^2)\}_{i=1}^N$ be a set of pairs of closest points for S_1 and S_2 . During the algorithm procedure the average distance between the point clouds S_1 and S_2 is minimized:

$$E = \frac{1}{N} \sum_{i=1}^N d(s_i^1, s_i^2) \rightarrow \min, \quad (1)$$

where $d(\cdot, \cdot)$ is the Euclidean distance between two points.

To calculate the distance between the point $s^1 \in S_1$ and the cloud of points S_2 in the original version of the algorithm [6] the "point-to-point" distance was used:

$$\rho(s^1, S_2) = \min_{s^2 \in S_2} d(s^2, s^1). \quad (2)$$

Metric (2) uses discrete surface models.

In [9] "point-to-plane" distance was proposed, the use of which implies that at each point of S_1 and S_2 normal to the surfaces specified by point clouds S_1 and S_2 , respectively, is known. Such information may be given initially, or calculated by averaging the normals of the adjacent triangles using triangulates polygonal approximation of the surface. In this case it the sum of squared distances between s_i^1 and plane P_i^2 , perpendicular to S_2 at s_i^2 over all pairs of nearest points (s_i^1, s_i^2) will be minimized:

$$E = \sum_{i=1}^N H^2(s_i^1, P_i^2) \rightarrow \min, \quad (3)$$

where $H(s, P)$ — Euclidean distance between s and plane P .

General iterative scheme of the algorithm is the following:

1. Search of pairs of points (s_i^1, s_i^2) , $i = \overline{1, N}$ for the current mutual position of S_1 and S_2 .
2. Search of the transformation (parameters of translations and rotations) of point cloud S_1 , reduces error E (1) and (3) using the least squares method. If the change of the error E is below a certain threshold, the algorithm terminates.
3. Application of the transformation found at the previous step to convert point cloud S_1 . Transition to Stage 1.

Stages 1 — 3 are repeated while the reduction of the error exceeds a certain threshold value. The result is the final position of point cloud S_1 .

Influence of "wrong" pairs of closest points is smaller for "point-to-plane" distance. But the disadvantage of the "point-to-plane" distance is that it is highly dependent on the initial relative position of S_1 and S_2 [10].

The main advantage of the ICP algorithm is its simple implementation, the disadvantage is a strong dependence on the initial approximation of objects, the computational complexity associated with search of all pairs of closest points of $\{(s_i^1, s_i^2)\}_{i=1}^N$.

Let N_1, N_2 be the numbers of points in initial point cloud S_1, S_2 , respectively. Then using a simple implementation the estimation of computational complexity of such search is $O(N_1 N_2)$, i.e. is quadratic if $N_1 \approx N_2$. Using more complex data structures — for example, k-d trees [11] — the search can be carried out in a time $O(N_1 \log N_2)$. Thus, the total number of operations required to find pairs of closest points in m iterations is $O(m N_1 \log N_2)$.

Large number of papers is devoted to various improvements of the algorithm:

- Modification of methods for selecting the region of overlap and pairs of closest points (for example, by introducing limitations to the class of motions that take one point to another [12], using the theory of random quantities [13], k-d trees [11], [7], genetic algorithms [14]);

- Introducing weights for pairs of closest points [15];

- Modification of the equation for the distance between two points [16], [17];

- Modification of the equation for the error to be minimized [18] and minimization procedure [17].

The drawback of the such improvements is their complexity, tuning to a specific experimental data, which reduces the robustness of the algorithms.

In [19] the problem of comparison of the surfaces for one and the same object, defined on different sets of points was considered. The distance from a point of one surface to the other surface was calculated along the normal to the second surface splines. Such measure is interesting because it does not require a reduction of the source data to common mesh. But the complexity of this approach is quadratic.

Such measures as, e.g., the average distance between the heights that can be directly calculated by recalculation of two surfaces at the nodes of common mesh, do not allow extensions to the case of their definition at two different meshes without the stage of transformation to regular meshes.

In [20] a surface matching algorithm based on minimizing the similarity measure between them was proposed. The compared surface S_1 and S_2 are considered as objects of linear space, and the similarity measure $\rho(S_1, S_2)$ is the norm $\|S_1 - S_2\|$ in this space. Let surface X be represented by triangulated piecewise linear model of N triangles, c_i — centroid of i -th triangle, n_i — normal vector to i -th triangle, the length of which is equal to the area of the triangle. The norm of surface X is introduced as

$$\|X\| = \sum_{i=1}^N \sum_{j=1}^N (n_i, n_j) e^{-|c_i - c_j|^2 / \sigma^2}. \quad (4)$$

Let the initial surfaces S_1, S_2 be represented by triangulated piecewise linear model of N_1, N_2 triangles, respectively; $c_i^1(c_i^2)$ — centroid of i -th triangle of surface $S_1(S_2)$, $n_i^1(n_i^2)$ — normal vector to i -th triangle of surface $S_1(S_2)$, whose length is equal to the area of the triangle. The similarity measure proposed in [20], is defined as follows:

$$\begin{aligned} \rho(S_1, S_2) = & \sum_{i=1}^{N_1} \sum_{j=1}^{N_1} (n_i^1, n_j^1) e^{-|c_i^1 - c_j^1|^2 / \sigma^2} + \sum_{i=1}^{N_2} \sum_{j=1}^{N_2} (n_i^2, n_j^2) e^{-|c_i^2 - c_j^2|^2 / \sigma^2} - \\ & - 2 \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (n_i^1, n_j^2) e^{-|c_i^1 - c_j^2|^2 / \sigma^2} = \|S_1 - S_2\|. \end{aligned} \quad (5)$$

Further, the value of the measure (5) is minimized by quickest descent method. The disadvantage of this approach is the quadratic complexity of similarity measure calculating.

3 Approach to Comparison of Point Clouds

The general formulation of the problem is the following: there exist two surfaces determined by the height functions at two finite sets of points; it is necessary to calculate some measure of similarity (or difference) between them.

Let T_1, T_2 , and T be the Delaunay triangulations [21] constructed on the sets of nodes of the meshes g_1 and g_2 and the common mesh $g = g_1 \cup g_2$, respectively. Then T is a *common*, or *united*, *Delaunay triangulation*. Let us denote by $Conv(g)$ the convex hull of mesh g . It is assumed that the initial meshes consist of non-intersecting sets of nodes.

3.1 Proposed Measures

A measure for comparing surfaces determined by single-valued functions at the nodes of different non-regular triangulation meshes was proposed in [22]. The sum of volumes of the difference between the functions over all triangles of T was calculated in order to determine the measure. In this case, the values of

the initial functions f_1 and f_2 represented the surfaces were interpolated at the points of the opposite mesh using the localization of the triangulations T_1 and T_2 in each other. As a result of interpolation, we obtain the continuous functions \hat{f}_1 and \hat{f}_2 determined on the set $Conv(g)$ for which $\hat{f}_1 = f_1$, $\hat{f}_2 = f_2$ at the nodes of g_1 , g_2 , respectively. Let $\mu(x, y) \geq 0$ be the function determining the weight of the difference of the surfaces at the point (x, y) . It is assumed that $\mu(x, y)$ is defined and finite at all points of $Conv(g)$ and is equal to zero beyond $Conv(g)$.

Let us introduce the weighted volume of the difference between the surfaces in the triangular region,

$$V_\mu(A, B, C, f_1, f_2) = \iint_{\Delta ABC} |\hat{f}_1(x, y) - \hat{f}_2(x, y)| \mu(x, y) dx dy. \quad (6)$$

For $\mu = 1$, the volume V_μ is the metric L_1 for the interpolated initial functions f_1 and f_2 on ΔABC . Let us denote by $S_{Conv(g)}$ the area of the convex hull of set g , which is equal to the sum of areas of all triangles of the triangulation of set g . The following measure was proposed for comparison of surfaces:

$$\rho_{V_\mu}(f_1, f_2) = \sum_{\Delta ABC \in T} V_\mu(A, B, C, f_1, f_2) / S_{Conv(g)}, \quad (7)$$

Summing in (7) takes place over all triangles in T .

The proposed measure can be adapted for each particular application by introducing the function μ .

In [23] special measure ρ_{dV} for case when two initial surface models have different level of detalization was proposed.

3.2 Surface Comparison Algorithm

An approach proposed for surface comparison and calculating measured is as follows: Delaunay triangulations are constructed on both meshes, then each of the functions is interpolated with respect to the other mesh, which is followed by the construction of the common triangulation for the two meshes. Then at each point of the merged meshes the values of the two functions are known, and operations can be performed on individual faces (triangles) of the common triangulation analyzing the mutual arrangement of the spatial triangles defined by the functions. Therefore, the developed algorithm is based on the idea of supplementing the values of each function in the nodes of the other mesh by constructing the Delaunay triangulation and their localization in each other.

Let us present main stages of the algorithm A^ρ for calculating measure ρ :

1. Construction of Delaunay triangulations T_1 , T_2 ;
2. Construction of minimal spanning trees of Delaunay triangulations;
3. Localization of each of meshes g_1 , g_2 in the triangulation constructed on the other mesh;
4. Interpolation of each of two functions f_1 , f_2 on the mesh that the other function defined on;

5. Construction of the common triangulation T (see the linear merging method proposed in [24]);
6. Comparison of the functions on the separate cells of triangulation T (calculating measure ρ).

3.3 Computational Complexity

The numerical complexity of algorithm A^ρ is $O(n \log n)$, where n is the total number of points of mesh g . It was proved that in case of Delaunay triangulations T_1 and T_2 are constructed at the preprocessing stage, the *expected time* to compute measure ρ_{V_μ} is $O(n)$. At the same time in the mentioned conditions the numerical complexity of calculating measure ρ_{dV} is $O(n)$ *in the worst case* (see the proof in [23]). The key stage of the algorithm that influenced on running time is the stage of mesh localization. The problem of computational complexity of stage 3 will be considered in Section 4.

3.4 Surface Matching

The numerical complexity of one calculation of the measure is important because in applications it is required to calculate the measure several times during iteration process of surface matching. Influence of non-efficient calculation of the measure will increase on each iteration step.

The matching problem, or the problem of spatial alignment of surfaces, consists in transition of several images of the considered object to one global coordinate system (see Fig. 2). Such transition consists in transformation of images using translations along and rotations about the coordinate axes. If we consider transformation parameters that do not derive Delaunay triangulations of the initial meshes from the class of triangulations, the proposed approach allows to implement surface matching in time $O(mn)$ using m iterations. In this case we need $O(n \log n)$ time for preprocessing stage to construct Delaunay triangulations.

4 Mesh Localization in Delaunay Triangulation

The well-known geometric search problem of point location in a Delaunay triangulation is formulated as follows: given a point Q and a Delaunay triangulation T , it is required to declare the triangle of T containing Q . One of the most fast methods to solve this problem (in general case) has $O(\log N)$ computational complexity and $O(N)$ memory usage, where N is the number of nodes in T [25].

Let g be a plane mesh of N_1 nodes and T_2 be a Delaunay triangulation constructed on N_2 nodes. In the problem of mesh nodes localization in a Delaunay triangulation it is required to solve point location problem for each node of g . Then the unstructured mass query of N_1 nodes can be processed by time $O(N_1 \log N_2)$.

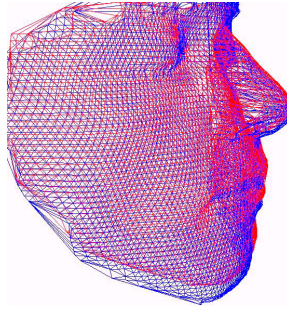


Fig. 2. Matching of facial models

We show how the Delaunay triangulation constructed on the nodes of g can be used for acquisition of more efficient solution. The proposed method is based on *Euclidean minimum spanning trees* (MST) of given Delaunay triangulation [21] which can be constructed in linear time. There is approach for point location problem that uses "walk along a line" strategy [26]. The idea of the approach consists in gradual transition from some initial point M of known location to source point Q along the straight line (MQ) . During each transition step changing on adjacent (neighboring by edge) triangle is implemented. After such stage is finished, a *location path* consisting of adjacent triangulation triangles is constructed. Case of belonging of a certain node of T_2 to segment $[MQ]$ is a case of a special interest.

In the proposed method for mesh localization, locations paths pass along the edges of the MST. As a spanning tree does not contain cycles and passes through all points of the mesh g , the algorithm will work correctly: it does not loop and performs location of all mesh nodes.

In this work we prove that in case of uniform distribution for nodes of triangulation T_2 and mesh g the average case complexity of the method is linear by $\max(N_1, N_2)$. Using the results of Kostuk for average edge length in Delaunay triangulation [27] and of Bose and Devroye for average number of intersections between MST and line segment [28], we show that

Lemma 1. *Suppose g_1, g_2 are plane meshes with the numbers of nodes N_1, N_2 , respectively, $N_1/N_2 \leq c = \text{const}$, and the sets of nodes of the both meshes uniformly distributed uniformly in a rectangular area. Then, the average number of intersections between the MST for nodes of g_1 and edges of the Delaunay triangulation constructed on g_2 is linear by N_2 .*

Theorem 1. *Under the conditions of lemma 1, the algorithm for localization of mesh g_1 in the Delaunay triangulation constructed on the set of nodes of g_2 , on basis of the MST of g_1 has the average case complexity $O(\max(N_1, N_2))$.*

The proof of lemma 1 and theorem 1 is omitted.

The assumption of uniform distribution of mesh nodes is appropriate for the majority of practical applications.

In the worst case the mesh localization is not linear. We consider method for construction of simulated example shows that the worst case complexity of the proposed method is quadratic.

In [24] the problem of merging of unseparated Delaunay triangulations were studied: given two Delaunay triangulations T_1 and T_2 constructed on sets g_1 and g_2 , respectively, with intersected convex hulls, it is required to construct the *united Delaunay triangulation* T . We say that a triangle of T is an *interface* triangle if it joins nodes from the both sets g_1, g_2 .

By N denote the total number of nodes in the sets g_1, g_2 . We consider the problem of identifying of all interface triangles of T . Using the idea proposed by Mestetskiy and Tsarik for merging of overlapping Delaunay triangulations in [24], we propose the solution of this problem and show that

Theorem 2. *All interface triangles of T can be extracted in linear time by N .*

The theorem 2 allows to receive the following result:

Theorem 3. *Localization of nodes from g_1 in a Delaunay triangulation constructed on g_2 on basis of the list of all interface triangles can be performed in $O(N)$ in the worst case.*

The theorems 1 and 3 were theoretically proved and experimentally verified. Experiments for estimation of computational complexity were performed on model (see Fig. 3) and real data.

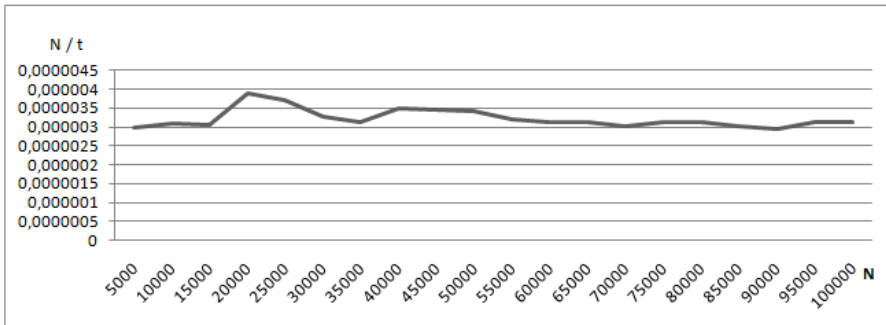


Fig. 3. Graph $N/t — N$, where $N \approx N_1 \approx N_2$ — the number of nodes in initial meshes, t — time for mutual localization of nodes of two Delaunay triangulations

5 Applied Problems of 3D Face Image Analysis

As the applications of the proposed methods several problems of 3D face analysis were considered:

- quantitative estimation of facial asymmetry based on comparison of the facial model and the reflected one [29];
- 3D model segmentation on static and dynamic regions by 3D model video sequence on example of chewing process [31]. The idea of estimating the motion parameters of the lower jaw consists in the conditional subdivision of each model of the video sequence into the upper (static) and lower (dynamic) parts and construction for each of the parts of special local coordinate systems of its own. The transformation of the lower system of coordinates into the upper one for every frame of the video sequence describes the dynamics of the lower jaw motion. The formal description of the motion is represented in the form of matrices of the lower coordinates transformation into the upper ones;
- construction of combined spatial model of face and jaws for orthodontics using *reference object*, i.e. object that allows to define geometric relationship of other objects connected with it [32];
- accuracy estimation of 3D model reconstruction methods [33];
- comparison of facial models acquired by scanners of different accuracy [23];
- quantitative estimation of facial skin condition for research in cosmetology [30].

The initial data for mentioned applications were acquired by 3D scanner Broadway designed by Artec Group Company (see [34]). Each face model was normalized in the coordinate system in a special way (transition to a standard coordinate system of a model). Facial models were considered as single-valued surfaces.

The experiments performed on facial model database confirmed the correctness and computational efficiency of the proposed methods.

6 Conclusions

Measures for comparing discrete models of surfaces determined at the nodes of different triangulation meshes and the approach for their calculation have been proposed. Several problems of face model analysis were considered as applications of the proposed methods. The proposed approach was theoretically justified and confirmed by multiple computational experiments on 3D face data.

References

1. Grise, G., Meyer-Hermann, M.: Surface reconstruction using Delaunay triangulation for applications in life sciences. *Computer Physics Communications* 182(4), 967–977 (2011)
2. Szymczak, A., Rossignac, J., King, D.: Piecewise regular meshes: Construction and compression. *Graphical Models* 64(3-4), 183–198 (2002)
3. Gu, X., Gortler, S.J., Hoppe, H.: Geometry images. In: *SIGGRAPH Conference Proceedings*, pp. 355–361 (2002)

4. Alliez, P., Ucelli, G., Gotsman, C., Attene, M.: Recent Advances in Remeshing of Surfaces. In: *Shape Analysis and Structuring: Mathematics and Visualization*, pp. 53–82 (2008)
5. Cignoni, P., Rocchini, C., Scopigno, R.: Metro: measuring error on simplified surfaces. *Computer Graphics Forum* 17(2), 167–174 (1998)
6. Besl, P., McKay, H.: A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14(2), 239–256 (1992)
7. Zhang, Z.: Iterative point matching for registration of freeform curves and surfaces. *International Journal of Computer Vision* 13(2), 119–152 (1994)
8. Fan, T., Medioni, G., Nevatia, R.: Recognizing 3D objects using surface descriptions. *IEEE PAMI* 11(11), 1140–1157 (1989)
9. Chen, Y., Medioni, G.: Object modelling by registration of multiple range images. *Image and Vision Computing* 10(3), 145–155 (1992)
10. Gelfand, N., Ikemoto, L., Rusinkiewicz, S., Levoy, M.: Geometrically Stable Sampling for the ICP Algorithm. In: *Fourth International Conference on 3D Digital Imaging and Modeling*, pp. 260–267 (2003)
11. Friedman, J.H., Bentley, J.L., Finkel, R.A.: An Algorithm for Finding Best Matches in Logarithmic Expected Time. *ACM Transactions on Mathematical Software* 3(3), 209–226 (1977)
12. Liu, Y., Rodrigues, M.A.: Geometrical analysis of two sets of 3D correspondence data patterns for the registration of free-form shapes. *J. Int. and Rob. Systems* 33, 409–436 (2002)
13. Clarkson, K.: A randomized algorithm for closest point queries. *SIAM J. Computing* 17, 830–847 (1998)
14. Brunnstrom, K., Stoddart, A.J.: Genetic algorithms for free-form surface matching. In: *Proc. ICPR*, pp. 689–693 (1996)
15. Godin, G., Rioux, M., Baribeau, R.: Three-dimensional registration using range and intensity information. In: *Proceedings of the SPIE*, vol. 2350, pp. 279–290 (1994)
16. Feldmar, J., Ayache, N., Betting, F.: D-2D projective registration of free-form curves and surfaces. *CVIU* 65, 403–424 (1997)
17. Turk, G., Levoy, M.: Zippered polygon meshes from range images. In: *Proc. SIG-GRAPH*, pp. 311–318 (1994)
18. Eggert, D.W., Larusso, A., Fisher, R.B.: Estimating 3-D rigid body transformations: a comparison of four major algorithms. *Machine Vision and Applications* 9(5-6), 272–290 (1997)
19. Schenk, T.: *Digital Photogrammetry*, 428 p. Terra-Science, Laurelville (1999)
20. Vaillant, M., Glaunès, J.: Surface Matching via Currents. In: Christensen, G.E., Sonka, M. (eds.) *IPMI 2005*. LNCS, vol. 3565, pp. 381–392. Springer, Heidelberg (2005)
21. Preparata, F., Shamos, M.: *Computational geometry. An introduction*, 390 p. Springer-Verlag GmbH (1985)
22. Dyshkant, N.: Measures for comparing discrete models of single-valued surfaces. *Moscow University Computational Mathematics and Cybernetics* 35(4), 193–200 (2011)
23. Dyshkant, N.: Measures for Surface Comparison on Unstructured Grids with Different Density. In: Debled-Rennesson, I., Domenjoud, E., Kerautret, B., Even, P. (eds.) *DGCI 2011*. LNCS, vol. 6607, pp. 501–512. Springer, Heidelberg (2011)
24. Mestetskiy, L., Tsarik, E.: Delaunay triangulation: recursion without spatial separation. In: *GrafiCon 2004: Proc. 14th Int. Conf., CMC MSU, Moscow*, pp. 267–270 (2004) (in Russian)

25. Kirkpatrick, D.G.: Optimal search in planar subdivisions. *SIAM J. Comput.* 12(1), 28–35 (1983)
26. Devillers, O., Pion, S., Teillaud, M.: Walking in a triangulation. *Internat. J. Found. Comput. Sci.* 13, 181–199 (2002)
27. Kostuk, Y.L.: Graphic search with the use of triangulation and cellular partition. *Vestnik of Tomsk State University* (275), 147–152 (2002) (in Russian)
28. Bose, P., Devroye, L.: Intersections with Random Geometric Objects. *Computational Geometry: Theory and Applications* 10(3), 139–154 (1998)
29. Dyshkant, N., Mestetskiy, L.: Estimation of Asymmetry in 3D Face Models. In: *International Conference on Computer Vision Theory and Applications (VISAPP 2009)*, pp. 402–405. INSTICC Press, Lisbon (2009)
30. Artuchin, S.: Quantitative Estimation of Skin Condition Based on 3D Face Scanning. In: *18th International Student, Postgraduate and Young Scientist Conference "Lomonosov-2011"*. Maks Press (2011) (in Russian)
31. Gordeev, D., Dyshkant, N.: Construction of Jaw Movement Model During Chewing Process by 3D image sequence. In: *GrafiCon 2009, Proc. 19th Int. Conf. Mosk. Gos. Univ., Moscow*, pp. 348–352 (2009) (in Russian)
32. Dzaraev, C., Persin, L., Porochin, A.: Using of 3D Scanners for Diagnostics of Dentoalveolar Anomalies. In: *Theoretical and Practical Forum Dental-Review, Moscow* (2010) (in Russian)
33. Mestetskiy, L., Dyshkant, N., Gordeev, D., Kumari Sharmila, M., Shekar, B.H.: Surface measures for accuracy evaluation in 3D face reconstruction. *Pattern Recognition* 45(10), 3583–3589 (2012)
34. Artec Group Inc. 3D Scanning Technologies, <http://www.artec3d.com>