

Similarity analysis using Hausdorff metrics

Aleksandr Mezhenin^[0000-0002-7150-9811], Alena Zhigalova^[0000-0001-7076-0951]

ITMO University, Kronverksky Ave. 49, St. Petersburg, Russia
a.v.mezhenin@gmail.com, alenazhigalova@mail.ru

Abstract. In this paper we present aspects of similarity analysis of polygonal models of arbitrary topological type. To arrive at such estimates, general topology metrics was used, more specifically, the Hausdorff dimensionality. Calculation of the determined three-dimensional polygonal models metrics will be subjected to analyses. The aspects of drawing normal vectors to the surfaces under examination, that are crucial to the calculation of the Hausdorff dimensionality, has been considered and analyzed. To improve the precision of metric calculation at the stage of construction of normal vectors to the surface, it is suggested to employ the averaging method - the calculation of the weighted average between the normal vectors of the neighboring triangles.

Since there exists an issue of accuracy when Hausdorff metrics calculation is performed, to improve the precision of metric calculation at the stage of construction of normal vectors to the surface, it is suggested to employ the averaging method - the calculation of the weighted average between the normal vectors of the neighboring triangles.

Keywords: Hausdorff metric, Virtual simulators, 3D reconstruction, Photogrammetry.

1 Introduction

The main purpose of video surveillance systems is to provide a visual inspection of an object equipped with them. When designing, it is necessary to take into account the combination of factors that increase the efficiency of the developed video surveillance systems. One of the most important factors is the coverage zone of the observed area with cameras and the absence of so-called "blind zones". Their appearance depends on many factors: the characteristics of the cameras being installed, their location, the number, used data formats [11].

Obviously, the more cameras, the easier it is to place them so as to get the most complete coverage of the observed scene, but this increases the cost of the developed system. The proposed computer simulation of the coverage area in the form of a cloud of points of the observed surfaces of the 3D space, according to the authors, will allow obtaining preliminary estimates of the effectiveness of the developed video surveillance systems [16] at the design stage, which will improve design solutions. A mathematical model is proposed, as well as algorithms for modeling a cloud of points and obtaining heat maps for comparing coverage areas [9].

2 Analog overview

Ledas Geometry Comparison (LGC) helps to inter compare arbitral 3D models and their structural assemblies. The result of the LGC comparison is thus presented in a user-friendly way, indicating groups of differing. Moreover, the LGC approach confronts each other identical faces of two models, restoring associative and allowing easy surface planes-related information transfer from one model to another: attributes, parameters, assembly issues, etc. If the position of one of the models has been changed by shifting or turning, the LGC returns the models to the same position, and performs a more consistent comparison of the geometry in this position. LGC approach does not depend on a specific 3D data format. For the present, LGC supports majorin dependent formats.

MESHLAB - open source solution allowing to compare polygon meshes. The program implements a Hausdorff distance filter, which calculates the distance from the grid X to Y. First and foremost, the Hausdorff metric between two meshes is the maximum between two so-called one-sided Hausdorff distances (technically speaking, it is not distance). These two measures are not symmetrical (for example, the results depend on which mesh is given as X). The Hausdorff MeshLab filters enable the user to calculate only a one-sided version. A sample based on the ensemble of X mesh points is used for calculation while for each x the nearest point y on the grid Y is recognized. This means that the result depends heavily on how many points on X are taken. A general approach is to use the mesh apex with the highest density as sample points for this purpose we select the “Apex Sampling” option in the dialog box. It is important to become certain that the number of samples is greater than or equal to the number of apices. The gathered information is recorded in the layers log window. This project currently exists as a web application.

3 Determining the distance between triangles on the plane

Let us consider a simplified version of the problem for the 2D case of calibration of distance between plane triangles. It is essential that we find the closest of the three possible triangles to the given one. For this purpose the centers of the triangles and the crosscut of bisectors. Then the distances to the centers of other triangles are calculated, and the closest one is recognized. This algorithm is set out below and was carried out in the Matlab environment.

Problem: there is a triangle, we need to recognize the closest to its center of three other triangles.

The scheme of the algorithm:

1. ΔABC - initial. The coordinates of its three points are keyed in.(a, b, c in Figure 5).
2. The coordinates of the points of the three other triangles are keyed in.
3. The in center ΔABC , the bisectors crosscut point O, is calculated:

$$O_x = \frac{a * x_A + b * X_b + c * X_c}{a + b + c} \quad (1)$$

where a; b; c - sides of a triangle, Xi -X-direction coordinate of the point.

$$O_y = \frac{a * Y_A + b * Y_b + c * Y_c}{a + b + c} \quad (2)$$

4. Calculation of the minimum distance from point O to three triangles, consider at the example of one of them - $\Delta P1P2P3$. For each point P1P2P3, the distance Pi from point O is calculated and compared with the previous Pi..1, as a result, the minimum OP is selected. The distance is estimated through the built-in PDIST Matlab feature for the Euclidean distance calculation [11]. It is worth noting that this feature is also available for the dimensionality space 3. $d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$ distance between points x and y in n-dimensional space.

5. As of the current step, there are 3 distance values (disto1, disto2, disto3 in Figure 5) from O point to the triangles. The maximum and minimum values among them are calculated (distomax, and distomin in Fig. 5).

6. In this manner, the triangle with the maximum distance, the one, which is further from the point O, changes to a red color. The nearest - neighbor turns green. The rest turns yellow.

4 Determining the distance between triangles on the 3D

For the sake of simplicity, let us imagine discrete 3D models represented by triangular meshes, since this is the most general way of representation of such data. The triangular mesh M will be a representation of the ensemble of points P in R^3 (apices) and the ensemble of triangles T that describe the connection between the apices of P .

Let us denote the two continuous surfaces S and S' , and

$$d(p, S') = \min_{p' \in S'} \|p - p'\|_2, \quad (3)$$

where $\|\cdot\|_2$ - is the Euclidian norm.

Therefore Hausdorff metrics between S and S' : $d(S, S') = \max_{p \in S} d(S', S)$.

It is important to understand the fact that the metrics is not symmetrical, h.e. $d(S, S') \neq d(S', S)$. Let us denote $d(S, S')$ as the direct distance, $d(S', S)$ as inverse distance. Then the symmetrical metrics:

$$d_2(S, S') = \max[d(S, S'), d(S', S)]. \quad (4)$$

Symmetric metrics ensures a more precise measurement of an error between two surfaces, since the calculation of a “one-way” error can lead to significantly underestimated distance values, as it was shown in Figure 1.

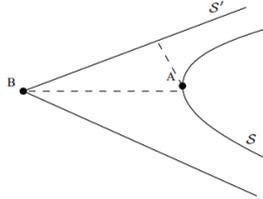


Fig. 1. Distance Comparison.

One can see that $d(S, S')$ is smaller than $d(S', S)$, since $d(A, S') \ll d(S, B)$. Thus, a not very large one-way distance does not mean a small presentation. The calculation of the Hausdorff distance between two discrete surfaces $M(P, T)$ and $M'(P', T')$ is related to the preceding definitions. Let us focus on calculation of the Hausdorff direct distance, h.e. $d(M, M')$, since the symmetric distance can be calculated from the direct and inverse distances. The distance between any point p from M (p is assumed not to be from P) and M' can be calculated from the estimation of the distance minimum between p and all triangles $T \in T'$.

When the orthographical projection p' of p on the plane T' is inside the triangle, the distance between the point and the triangle is simply a distance from the point to the plane. When the projection remains outside T' , the distance between the point and the plane is the distance between p and the closest p'' from T' , which should lie on one of the sides of T' (Fig. 2).

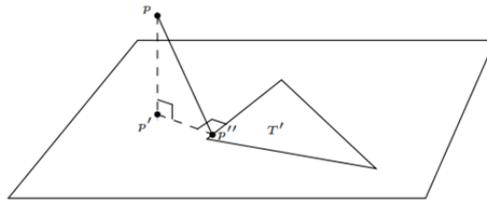


Fig. 2. Projection Construction.

Although $d(p, S')$ can be calculated for any point p , it is essential to perform sampling to calculate the maximum $p \in S$. Each T triangle is sampled, and the distance between each sample and M' is estimated. Each triangle sampling is performed as follows: two sides of the triangle are considered as directions for the sample lattice. In

accordance with the criterion of length, each side is selected with n points. By means of directions, it is easy to construct a 'correct' mesh of the triangle under consideration (Fig. 3). According to this sampling, $n(n+1)/2$ samples are constructed in each triangle. An interesting property of this sample is that the triangle can be split into smaller ones that possess all the same areas, which leads to much simpler calculations of the integrals taken through surface.

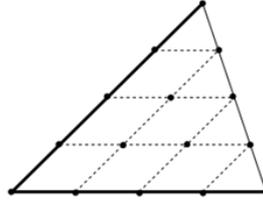


Fig. 3. Sampling for $n=5$.

Representative illustration of a sample made on a triangle for $n = 5$. The sides adopted as main directions are in bold and the samples are specified with black dots.

5 Mean Absolute and Quadratic Mean Errors

The distance from the point to the surface, defined in (1), can be used to determine the average error d_m between two surfaces S and S' :

$$d_m(S, S') = \frac{1}{|S|} \iint_{p \in S} d(p, S') dS, \quad (5)$$

where $|S|$ stands for the S area.

Then we get the definition of the quadratic mean error:

$$D_{mse}(S, S') = \sqrt{\frac{1}{|S|} \iint_{p \in S'} d(p, S')^2 dS} \quad (6)$$

By means of an equation (2), symmetric variants of the mean absolute and quadratic mean error can be determined.

The calculation of such values for discrete models is quite simple, if the error values (see Figure 3) can be calculated for each example.

The (quadratic) error integral over the entire surface is calculated by summing the contributions of all parallelograms formed by 4 samples (see Fig. 3), plus the marginal triangle. Let us denote four samples inside the triangle by $x_{i,j}$, $x_{i+1,j}$, $x_{i,j+1}$, $x_{i+1,j+1}$ and error value associated with each one by $e_{i,j}$, $e_{i+1,j}$, $e_{i,j+1}$, $e_{i+1,j+1}$.

The e integral which is formed by samples x over the parallelogram can be cut into two triangles, as one can see in Figure 4. Let us now focus on the e integral over the triangle $T_{i,j} = (x_{i,j}, x_{i+1,j}, x_{i,j+1})$.

The simplest method of the error integral calculation involves linear interpolation between the values of $e_{i,j}$, $e_{i+1,j}$ and $e_{i,j+1}$. The concept of the sampling method means that within each triangle from T , the samples are easily triangulated (shaped into triangles), as shown in fig. 4, and all the resulting triangles have equal area.

The value of the integral is then $|T_{i,j}| \cdot (e_{i,j} + e_{i+1,j} + e_{i,j+1})/3$. The e^2 integral for the same triangle is also calculated via linear interpolations between e values (resulting in quadratic interpolation between e values) and, finally, the integral value:

$$|T_{i,j}| \cdot [e_{i,j}(e_{i,j} + e_{i+1,j} + e_{i,j+1}) + e_{i+1,j}(e_{i+1,j} + e_{i,j+1}) + e_{i,j+1}^2] / 6 \quad (7)$$

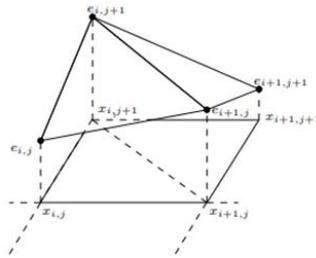


Fig. 4. Integral Construction.

6 Experimental results

The proposed method of comparison of polygonal objects based on the calculation of Hausdorff dimension was implemented in MATLAB ImageProcessingToolbox (IPT) as m-functions, as a result of which the calculated normal vectors and the value of deviations of one surface from another are visualized. For Fig. 5 a showing 2D case of calibration of distance between plane triangles.

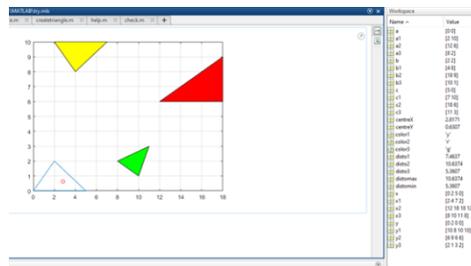


Fig. 5. 2D case of calibration of distance between plane triangles.

For Fig. 6 a graph showing the use of the developed m-functions for comparison of different polygonal surfaces, which are the result of two mathematical three-dimensional functions, showing the normal vectors, constructed from polygons of one

surface to another, and the magnitude of the deviation is presented in the form of a graph color palette.

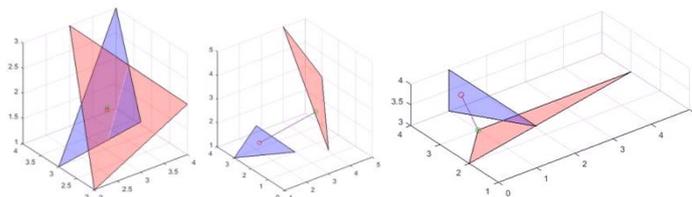


Fig. 6. Construction of normal vectors to different surfaces.

7 Conclusion

Studies have shown that virtual environments can be used to verify the application of intelligent systems, debugging and testing of photogrammetric systems, machine vision, etc. Synthetic image generated mathematically equivalent to the actual photographs. Thus they may be used as a calibration image.

For a more accurate assessment of the efficiency and quality of these approaches is planned to conduct additional testing these methods simulation of different shooting conditions: the conditions of uneven lighting, noise, taking into account the reflection properties of the objects materials.

The proposed method of analysis of the similarity of polygonal models of arbitrary topological type can serve as a basis for the implementation of appropriate algorithms. Using the weighted average in calculating the normal vectors, according to the authors, further increases the accuracy of calculating Hausdorff metric. The proposed approach can be applied to problems of assessing the quality of 3D reconstruction algorithms and pattern recognition models, as well as problems of levels of detail representation of polygonal models.

Acknowledgments. The research has been supported by the RFBR, according to the research projects No. 17-07-00700 A.

References

1. Dyshkant, N.: Measures for Surface Comparison on Unstructured Grids with Different Density. Lecture Notes in Computer Science: Discrete Geometry for Computer Imagery, Vol. 6607, pp. 501–512 (2011).
2. Piatti, E., Lerma, J.: A Virtual Simulator for Photogrammetry. ISPRS Comm. V Symposium 25-27 September, Dresden (Germany), XXXVI (2006).

3. Klasing, K. Wollherr, D., Buss, M.: A clustering method for online segmentation of 3d laser data. In proceedings ICRA (2008).
4. Huang, J., Menq, C.: Automatic data segmentation for geometric feature extraction from unorganized 3-d coordinate points. In IEEE Trans. Robot. Automat., vol. 17, pp. 268-279 (2001).
5. Hielsberg, M., Tsai, R., Guo, P., Chen, C.: Visibility-based urban exploration and learning using point clouds. Under review (2013).
6. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Matching 3D Models with Shape Distributions. In proceedings of the International Conference on Shape Modeling & Applications, SMI '01, p. 154 (2001).
7. Ahmadian, H., Farughi, S.: Shape functions of superconvergent finite element models. Thin-Walled Structures 49, pp. 1178–1183 (2011).
8. Sizikov V.S., Stepanov A.V., Mezhenin A.V., Burlov D.I., Eksemplyarov R.A. Determining image-distortion parameters by spectral means when processing pictures of the earth's surface obtained from satellites and aircraft // Journal of Optical Technology. 2018, Vol. 85, No. 4, pp. 203–210.
9. Mezhenin A.V., Izvozchikova V.V.: Metody postroenija vektorov normalej v zadachah identifikacii ob#ektov. Kibernetika i programirovanie. 2013. № 4. S. 51-58.
10. Mezhenin A.V. Issledovanie sistem stereozrenija sredstvami virtual'nogo modelirovanija. V sbornike: Voprosy obrazovanija i nauki: teoreticheskij i metodicheskij aspekty 2012. S. 96-98.
11. Mezhenin A.V. Analiz podobija poligonal'nyh modelej v graficheskijh sistemah / A.V. Mezhenin, A.Ju.Krotova, A.L.Zelenkovskij // Informatika i vychislitel'naja tehnika: sbornik nauchnyh trudov / pod red. Vojta. – Ul'janovsk: UIGTU, 2011. -656 s., S. 397-400.
12. Mezhenin, A.V. Metody postroenija vektorov normalej v zadachah identifikacii ob#ektov / A.V.Mezhenin, V.V. Izvozchikova // Kibernetika i programirovanie. -2013. -№ 4. S. 51-58.
13. Mezhenin A.V. Analiz podobija poligonal'nyh modelej v graficheskijh sistemah / A.V. Mezhenin, A.Ju.Krotova, A.L.Zelenkovskij // Informatika i vychislitel'naja tehnika: sbornik nauchnyh trudov / pod red. Vojta. – Ul'janovsk: UIGTU, 2011. -656 s., S. 397-400.
14. Mezhenin, A.V. Issledovanie sistem stereozrenija sredstvami virtual'nogo modelirovanija / A.V.Mezhenin // Voprosy obrazovanija i nauki: teoreticheskij i metodicheskij aspekty. Sb. nauch. tr. po mat-lam Mezhdunar. zaoch. nauch.-prakt. konf. 30 aprilja 2012 g. - Tambov: Min. obrazovanija i nauki Ros. Federacii, Chast' 4. - S. 96-98. - 164 s (2012).
15. Mezhenin, A.V. Rekonstrukcija trehmernyh modelej po rastrovym izobrazhenijam / A.V.Mezhenin, V.T.Tozik // Nauchno-tehnicheskij vestnik informacionnyh tehnologij, mehaniki i optiki. -№45. S.203-207, (2007).
16. Shardakov, V.M., Parfenov, D.I., Zaporozhko, V.V., Izvozchikova, V.V. Development of an Adaptive Module for Visualization of the Surrounding Space for Cloud Educational (2018).