

# Point Cloud Registration Hybrid Method

Aleksandr Mezhenin<sup>a</sup>, Vera Izvozhikova<sup>b</sup>, Anna Grigoreva<sup>a</sup> and Vladimir Shardakov<sup>b</sup>

<sup>a</sup> *University ITMO University, Kronversky ave. 49, St. Petersburg, Russia*

<sup>b</sup> *Orenburg State University, ave. Pobedy 1, Orenburg, Russia*

## Abstract

Methods for registering point clouds based on a series of 2D images are considered. An overview of existing methods for registering point clouds is given: the iterative closest points (ICP) algorithm, its modification using instead of the Euclidean metric, the Hausdorff metric. To find the corresponding points, it is proposed to use the method of optimization of the normal vector and particle swarm (NVP). The article discusses software for photogrammetry DF Zephyr Free, Meshroom on the AliceVision platform, VisualSFM, and the algorithm proposed by the authors implemented on the MATLAB platform. Approaches to assess the accuracy of the resulting models and visual quality are proposed. Using Virtual Scenes for Comparison of Photogrammetry Software. Using heatmaps. The proposed solutions were tested.

## Keywords

Point Cloud Registration Algorithm, Image Preprocessing, Photogrammetric Software, Iterative Closest Point Algorithm, Hausdorff Metric, Particle Swarm Optimization, Normal Vectors, Heatmaps, Kullback-Leibler Divergence, Neighborhood Volume.

## 1. Introduction

Currently, various methods are used to synthesize a virtual environment, such as modeling in computer graphics programs and modeling using photogrammetry. However, the method of representing objects in space in the form of clouds of points of different densities is the most promising [1, 3]. Data for building a point cloud can be obtained by scanning 3D objects with special devices, as well as by processing optical scan data [4].

A number of commercial tools and open source software are currently available to handle all steps of an image-based 3D modeling framework from image preprocessing to 3D geospatial products such as point clouds, DSMs, orthomosaics, mesh models, texture models, etc. e. Since choosing the best solution for 3D modeling and mapping is an important issue in many projects, the performance of modern photogrammetric software is being evaluated in ongoing research [2]. Below is a brief overview of the tested photogrammetry software.

**3DF Zephyr Free** [5]. Has a user-friendly interface that allows those who are just getting started with photogrammetry to dive into it right away. There are many wizards built into the program to help guide users. These wizards will help both experienced and novice users choose the correct settings. Although this software is completely free to download, its use has some limitations. The free version can use a maximum of 50 images. This will, of course, limit the amount of detail that can be displayed.

**Meshroom** [6]. Open source photogrammetry software based on the AliceVision Photogrammetric Computer Vision platform. This software operates as part of a node-based workflow that brings all the steps together to create the final 3D model. Each node can be worked individually to get the final result. One of the great things about the software is that it will show the 3D model in real time as

---

Proceedings of the 12th Majorov International Conference on Software Engineering and Computer Systems, December 10-11, 2020, Online & Saint Petersburg, Russia

EMAIL: mejenin@mail.ru (A. 1); viza-8.11@mail.ru (A. 2); grishanya1998@yandex.ru (A. 3); werovulv@inbox.ru (A. 4)

ORCID: 0000-0002-7150-9811 (A. 1); 0000-0002-8707-9510 (A. 2); 0000-0002-7197-617X (A. 3); 0000-0001-6151-6236 (A. 4)



© 2020 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

improvements are made and images are added to the dataset. Over time, you will be able to see how additional data fills in gaps and incomplete parts of the model. During the final rendering of your 3D model, quite a lot of computer processor resources may be required, preferably a CUDA GPU. Photogrammetry results are generally quite good and the parameters are easy to adjust.

**VisualSFM** [7]. Photogrammetry software used to create 3D motion reconstruction. It is a free open source option that allows you to upload many photos that can be converted to 3D models. Although the instructions can be a little confusing, once you master them you will have more success. Better to start with small projects. By taking your time with the documentation, you can understand this program faster. One great thing about this software is that it can quickly create 3D renders in just a few minutes. You should also make sure that you have enough RAM on your system, as software can consume a little RAM.

The solution proposed by the authors for registering point clouds uses the Iterative Nearest Points Algorithm. Modified Hausdorff metric, Particle swarm method and implemented in MATLAB [8]. Three metrics as the image alignment, total spatial error, and reprojection error were used to compare the capability of both software in 3D modeling of complex structures.

## 2. Overview of Existing Methods for Registering Point Clouds

The existing point cloud registration methods can be divided into two types: one is the accurate positioning of the navigation system in the scanning process and the other is the accurate alignment of point clouds from different perspectives. The algorithm in this paper belongs to the second type. The most classical algorithms in the automatic registration of point cloud model are iterative closest point (ICP) and its improved algorithms. They are methods based on point-to-point or point-to-surface search technology, and point cloud registration is completed by minimizing the distance between point clouds. ICP is easy to implement, but it requires that there is a constraint between two point clouds, and the positions of the two point clouds are relatively close. The algorithm results depend on the initial position of the point cloud, which is easy to cause the problem of rapid convergence to local optimal. Therefore, there are a lot of works to do to improve the ICP algorithm [10].

### 2.1. Interactive Closest Points (ICP). Modified Hausdorff Distance (MHD)

The Interactive Closest Points (ICP) iterative algorithm is the most widely used method needed to minimize the distance between points. This algorithm is able to perform transformations by finding the rotation matrix  $R$  and the displacement vector  $T$ , which in turn align the point clouds  $X$  and  $Y$ . To translate each individual point of the cloud  $X$  into the coordinate system of the cloud  $Y$ , an orthogonal geometric transformation is used

$$y_i = Rx_i + T, \quad (1)$$

where  $T$  is the translation vector,  $R$  is the rotation matrix,  $x_i$  is an arbitrary point in the cloud  $X$ ,  $y_i$  is an arbitrary point in the cloud  $Y$ . The basic algorithm for determining the nearest points has a number of disadvantages, such as the need for a common overlapping area of point clouds, as well as the complexity of the operation of finding the nearest points at each iteration. To eliminate these shortcomings, several modifications of this algorithm were proposed, given below. The Generalized-ICP algorithm uses a probabilistic structure for determining the error function. This method takes points from solid areas and combines them, thereby improving the performance and accuracy of the ICP algorithm. The algorithm based on orientation histograms allows solving the problem without using initial approximations for the rotation matrix  $R$  and the transfer vector  $T$  (as specified in the basic ICP algorithm). In the course of the work, orientation histograms are obtained for the clouds of points  $X$  and  $Y$ . Then the maximum of the correlation function is found for all values of the rotation matrix, which is the desired angular orientation. The advantage of this modification lies in the efficiency of the algorithm for large initial angles between the input point clouds. To reduce the computational complexity, various metrics (instead of Euclidean) are also used to measure the distance between points. One such metric is the Manhattan metric (or distance of city blocks)

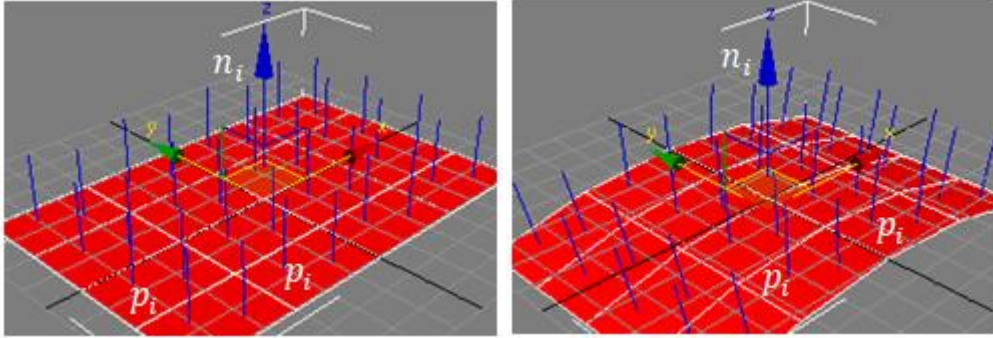
$$d(x, y) = \sum_{i=1}^n |x_i - y_i|. \quad (2)$$

The metric has better performance and noise immunity, less computational cost and less execution time compared to the base algorithm. Another used metric is the Hausdorff metric [9], which turns the set of all nonempty compact subsets of a metric space into a metric space

$$f = \max\{d(X, Y), d(Y, X)\}, d(X, Y) = \max_{x \in X} d(x, Y). \quad (3)$$

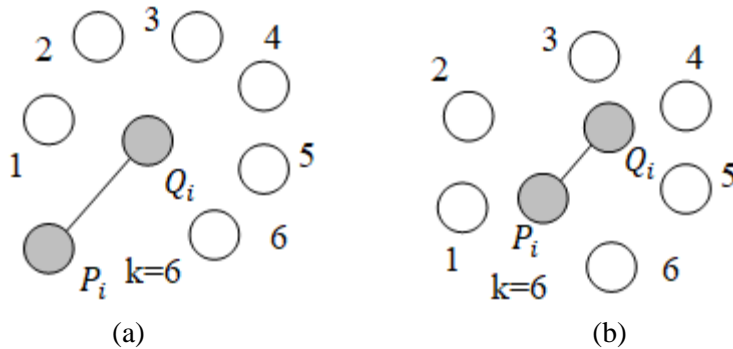
## 2.2. Particle Swarm Optimization

Based on normal vector and particle swarm optimization (PSO), a point cloud registration algorithm is proposed by searching the corresponding points [13]. It provides a new method for point cloud registration using feature point registration. One of the most important characteristics of a node in a point cloud is the normal vector. Accurate, high-quality point-based drawing methods not only depend on normal vectors, but accurate normal vectors are also required for accurate reconstruction results. The reconstruction algorithm is especially in need of normal vector aggregation such as multilevel division unit (MPU) [21], implicit surface reconstruction algorithm, and sharp feature detection and restoration. Methods for calculating the point cloud normal vector can be divided into three types. These are methods based on local surface approximation, Delaunay or Voronoi and reliable statistics. The calculation of the curvature of a point in this article belongs to the first type. The method was first proposed by Hoppe [15] when he was studying a surface reconstruction algorithm based on a signed distance function. Suppose that the plane of the point cloud is smooth everywhere (Figure 1). The steps for calculating the curvature are shown below.



**Figure 1:** Normal vectors from different surfaces: (a) the flat surface and (b) the non-flat surface

Suppose  $P = \{p_i(x_i, y_i, z_i) | i = 1, 2, 3, \dots, N\}$  is the input point cloud data and  $\{p_{ij}(x_{ij}, y_{ij}, z_{ij}) | j = 1, 2, 3, \dots, k\}$  is the  $k$ -nearest neighbors of point cloud  $p_i$ .



**Figure 2:** Different distances from the point to its neighbor center of gravity: (a) boundary point and its neighbor gravity center and (b) interior point and its neighbor gravity center

Then, the neighbor center of gravity is defined as follows

$$O_i = \frac{1}{k} \sum_{j=1}^k p_{ij} . \quad (4)$$

The covariance of  $p_i$  is calculated as follows

$$T_i = \begin{bmatrix} p_{i1} - O_i \\ p_{i2} - O_i \\ \dots \\ p_{ik} - O_i \end{bmatrix}^T \begin{bmatrix} p_{i1} - O_i \\ p_{i2} - O_i \\ \dots \\ p_{ik} - O_i \end{bmatrix} , \quad (5)$$

where  $T_i$  is a semi-positive symmetric matrix with geometric information of surface and solved by Jacobian method. The three eigenvalues ( $\lambda_1, \lambda_2, \lambda_3$ ) and their feature vectors ( $n_1, n_2, n_3$ ) are obtained. If  $\lambda_3 \geq \lambda_2 \geq \lambda_1$ , the point curvature  $C_i$  and the surface normal vector  $n_i$  are defined as follows

$$C_i = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3}, \quad n_i = n_{1i} . \quad (6)$$

By definition, it is shown that greater the change of surface is, the larger the point curvature is (see Figure 2).

### 3. Point Cloud Registration Suggested Solution

#### 3.1. Modification of the Basic ICP Algorithm. Calculating the Average Distance in the Hausdorff Metric

It is proposed to use the modified Hausdorff metric [10], which has the following form:

$$f = \max\{d(X, Y), d(Y, X)\}, \quad d(X, Y) = \frac{1}{N_x} \sum_{x \in X} d(x, Y). \quad (7)$$

The approach proposed by the authors consists in calculating the forward and backward distance between cloud points, and, in contrast to the usual Hausdorff metric, not the maximum, but the average value of the distance is calculated. To improve the accuracy of calculating the metric, it is proposed to use the averaging method - calculating the weighted average value of the normal vectors formed by pairs of adjacent triangles.

#### 3.2. Optimization of Normal Vectors and Particle Swarm

It Based on normal vector and particle swarm optimization (NVP), a point cloud registration algorithm is proposed by searching the corresponding points. It provides a new method for point cloud registration using feature point registration. First, in order to find the nearest eight neighbor nodes, the k-d tree is employed to build the relationship between points. Then, the normal vector and the distance between the point and the center gravity of eight neighbor points can be calculated. Second, the particle swarm optimization is used to search the corresponding points. There are two conditions to terminate the search in particle swarm optimization: one is that the normal vector of node in the original point cloud is the most similar to that in the target point cloud, and the other is that the distance between the point and the center gravity of eight neighbor points of node is the most similar to that in the target point cloud. Third, after obtaining the corresponding points, they are tested

by random sample consensus in order to obtain the right corresponding points. Fourth, the right corresponding points are registered by the quaternion method. The experiments demonstrate that this algorithm is effective. Even in the case of point cloud data lost, it also has high registration accuracy.

## 4. Assessment of the accuracy and visual quality of the results of a point cloud registration

### 4.1. Using virtual scenes to create synthetic datasets

As noted, the initial data for any photogrammetry system requires a series of overlapping photographs with certain properties, from which a three-dimensional reconstruction is subsequently obtained. To obtain a high-quality result, a certain number of photographic images are required. Shooting angle, camera parameters, placement of light sources are important [14]. For a preliminary assessment of the organization of the shooting process, the authors propose to use existing three-dimensional modeling systems, such as 3ds Max, Blender, etc. [18] (Figure 3). The quality assessment is supposed to be carried out by comparing test objects at the level of polygonal objects. This approach will allow not only to simulate shooting conditions, allowing you to get the best result. The use of virtual test scenes will allow, to a certain extent, to evaluate various photogrammetry systems. This can be used to improve existing software and develop new solutions.

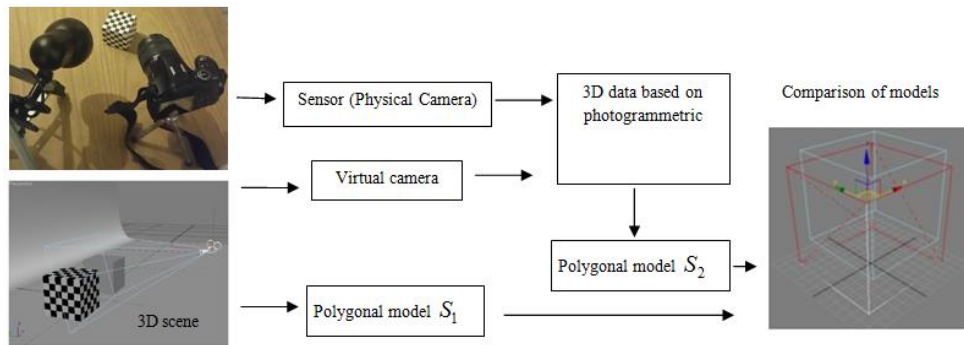


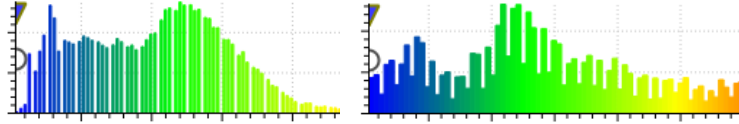
Figure 3: Pipeline simulation concept

The use of three-dimensional modeling systems will allow creating virtual stands of various configurations, simulating a different number of cameras, their position, stirring light sources (Figure 3).

### 4.2. Heatmaps as a tool for assessing the visual quality of the created scenes

Heatmaps are a popular visualization technique that encode 2D density distributions using color or brightness. To assess the visual quality of cloud registration, it is proposed to use heat maps. Initial data - images of point clouds obtained by different methods (systems). To do this, we calculate the difference between the values for each cell and determine the color based on the estimate of the resulting value, less than or greater than zero. In addition, we will use two different differences for calculations: a simple difference:  $D_i = B_i - A_i$  and a difference in ratios:  $D_i = B_i/A_i$ .

Heatmaps allow a visual assessment of the recorded point clouds. To obtain quantitative estimates, it is proposed to compare two heat maps. Comparing two heat maps is actually comparing two histograms or distributions (Figure 4). Thus, an alternative to the visual approach is to use statistical tools that calculate the difference between two distributions. To do this, you can use a functional - the Kullback-Leibler (DKL) divergence, a measure of the distance from each other of probability distributions [19].



**Figure 4:** Histograms, or distributions. DF Zephyr Free, Meshroom

Specifically, given a base distribution  $P$  and an approximation  $Q$ , the divergence is defined as

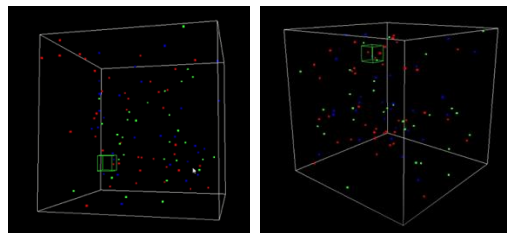
$$D_{KL} = \sum_{i=1}^n P(i) \log \frac{P(i)}{Q(i)} \quad (8)$$

First, for each possible value  $i$ , find the ratio of the probabilities to observe  $i$  under  $P$  and  $Q$ . Then take the log of this ratio, to find its order of magnitude. Next, weight these values by  $P(i)$ , meaning that more weight is assigned to the more probable values. Finally sum it all up to obtain a measure of the overall divergence. A small divergence signifies similar distributions, or a good model. Note that the first steps are the same as in our ratio difference: we assign colors based on the log of the ratio.

## 5. Experimental Results

### 5.1. Particle Swarm

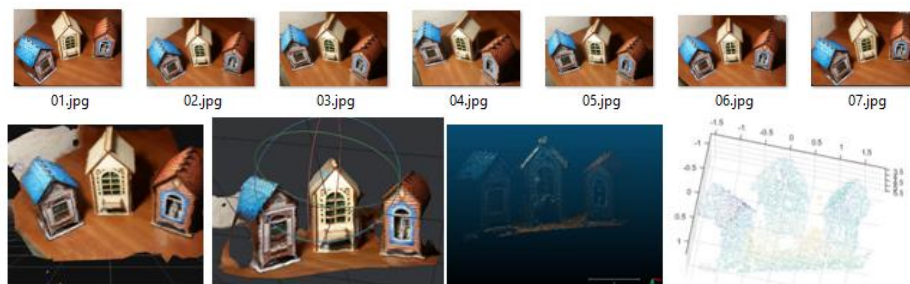
To test the algorithms for optimizing the normal vector and the swarm of particles, the authors have developed special software. Particle Swarm simulation results are shown in Figure 5.



**Figure 5:** Particle Swarm simulation

### 5.2. Point Cloud Estimates

The main stages of this research include: image acquisition, point cloud creation, accuracy and visual quality assessment. To compare the obtained point cloud, the software packages 3DF Zephyr Free, Meshroom, VisualSFM were taken. The ICP + NVP algorithm proposed by the authors was implemented on the MATLAB platform. The overall visual quality of the resulting 3D point cloud over various types of structural elements is shown in Figure 6 for each software.



**Figure 6:** DF Zephyr Free, Meshroom, VisualSFM. ICP+NVP (MATLAB)

The density output can be: the number of neighbors  $N$  (only available in 'Precise' mode); a surface density: number of neighbors divided by the neighborhood; a volume density: number of neighbors divided by the neighborhood:

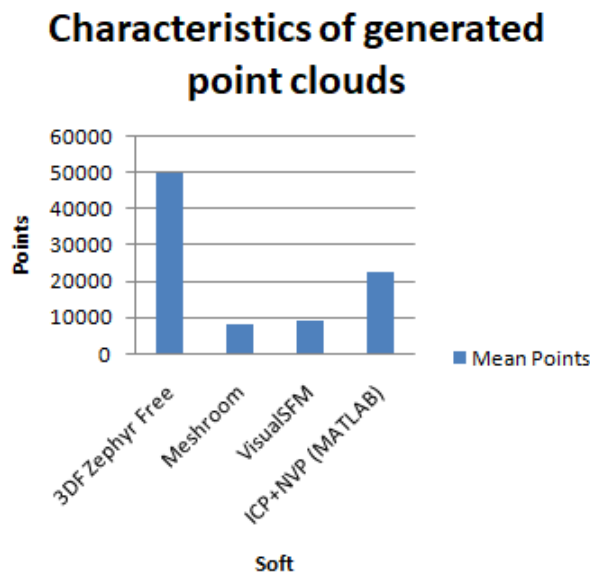
$$surface = N / (4/3 \pi R^2) \quad volume = N / (4/3 \pi R^3) \quad (9)$$

A point with no neighbor in the spherical neighborhood will have an invalid (NaN) density. The central point is always used for computing the density (even when the output is the 'number of neighbors' as we consider the number of neighbors of the 'position in space' of each point). Therefore the density will be actually equal to  $kNN + 1$  where  $kNN$  is the number of neighbors of each point. Point cloud parameters were obtained in the program CloudCompare [20] (Table 1). Received characteristics of generated point clouds by different software were used for further analysis. The number of points in the cloud obtained by different systems is shown in Figure 7.

**Table 1**

Characteristics of generated point clouds by different software

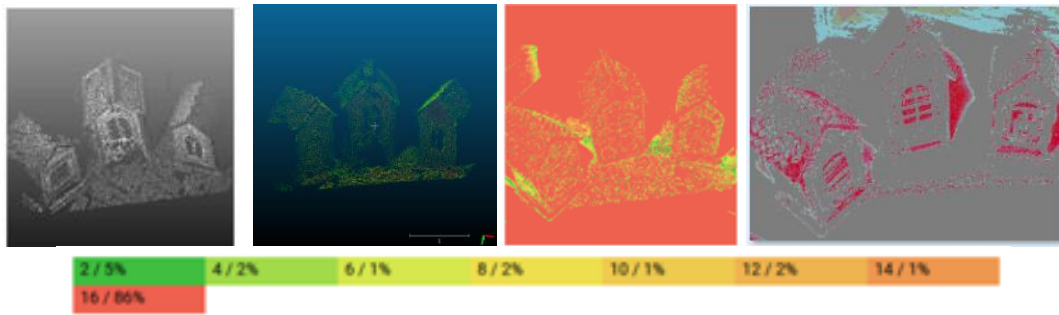
Software	Points	meanX	mean Y	mean Z	volume dens.	mean	std.dev.
Zephyr Free	49792	-0.654	3.825	6.828	0.326	89.6	45.5
Meshroom	8162	-0.039	-0.03	2.199	0.078	28457.3	16426.4
VisualSFM	9061	0.239	0.091	1.435	0.097	8632.8	2876.3
ICP+NVP	22338	0.307	1.315	3.487	0.164	12392.6	6449.2



**Figure 7:** Number of points in the cloud

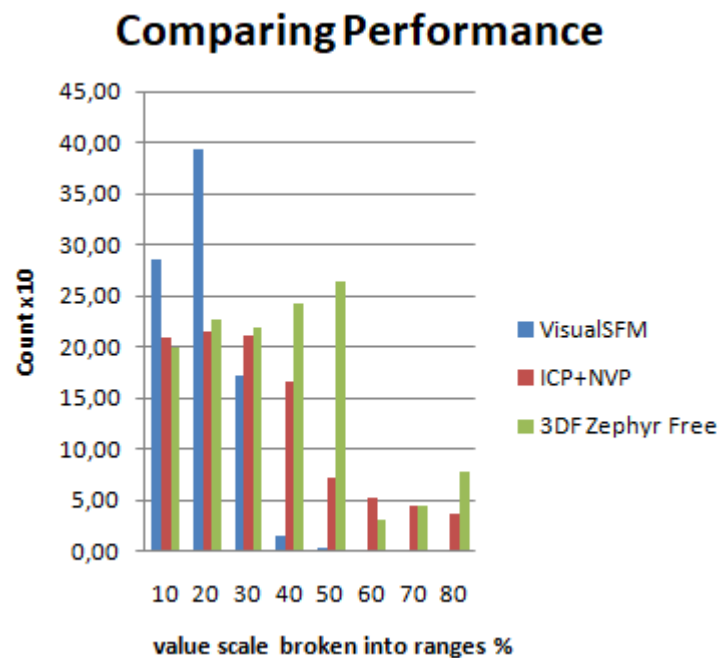
### 5.3. Point Cloud visual quality assessment using heatmaps

The next step is to compare the quality of building models based on the analysis of heat maps. A photo-image of the scene on the basis of which the cloud was built was used as a reference for obtaining heat maps. The heatmap shows deviations from the reference. Heatmaps are shown in Figure 8.



**Figure 8:** Heatmaps

The results of comparison of heat maps of the obtained clouds of the studied photogrammetry systems are presented in the form of a histogram (Figure 9). The count value on the histogram corresponds to the deviation from the standard. The assessment range is divided into 10 parts.



**Figure 9:** Comparing Performance

From the obtained histogram, it can be seen that the ICP + NVP algorithm as a whole shows less deviation from the standard.

## 6. Conclusion

It can be concluded that the differences in the total number of points for the four programs do not indicate significant quality differences in the final 3D reconstruction. For all software, due to various sources of errors, gaps are created during the matching process, and therefore the quality of the 3D reconstruction is reduced. For example, due to hidden areas and mismatch errors in complex structural areas, as well as shadows in the corners of the walls, the generated point clouds have gaps. However, it can be considered that these errors are small. In addition, there are large gaps in the walls due to the inappropriate viewing angle of images for all software. Also, the error of missing surface detail due to anti-aliasing processing is evident in the generated point cloud.

## 7. References



- [1] Knyaz, V.A., Zheltov, S.Yu.: Photogrammetric Techniques for Dentistry Analysis, Planning and Visualisation. In proceedings ISPRS Congress Beijing 2008, Proceedings of Commission V, pp. 783–788 (2008).
- [2] Alidoos F., Arefi H.: Comparison of UAS-based photogrammetry software for 3D point cloud generation. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume IV-4/W4, 2017.
- [3] Izvozchikova V., Shardakov V., Mezhenin A.: 3D Modeling Surveying Tasks in Photogrammetry. Proceedings - 2020 International Russian Automation Conference, RusAutoCon 2020, 2020, c. 220-224, 9208090.
- [4] Polyakov V., Mezhenin A.: Procedural Generation of Virtual Space. Advances in Intelligent Systems and Computing, 2020, 1156 AISC, c. 623-632.
- [5] 3DF Zephyr Free - a complete and free photogrammetry software. URL: <https://www.3dflow.net/3df-zephyr-free/>.
- [6] AliceVision | Photogrammetric Computer Vision Framework. URL: <https://alicevision.org/#meshroom>.
- [7] VisualSFM : A Visual Structure from Motion System. URL: <http://ccwu.me/vsfm/>.
- [8] MathWorks - Makers of MATLAB and Simulink. URL: <https://www.mathworks.com/>.
- [9] Marie-Pierre Dubuisson and Anil K. Jain: A Modified Hausdorff Distance for Object Matching. Proc. International Conference on Pattern Recognition, Jerusalem, Israel, pp 566-568, 1994.
- [10] Arunava Seal, Abhijit Bhowmick: Performance Analysis of Iterative Closest Point (ICP) Algorithm using Modified Hausdorff Distance. International Research Journal of Engineering and Technology (IRJET), Volume: 04 Issue: 07 | July -2017.
- [11] Mezhenin, A., Zhigalova, A.: Similarity analysis using Hausdorff metrics. CEUR Work-shop Proceedings, 2019. Vol-2344.
- [12] Mezhenin A., Izvozchikova V., Shardakov V., Korotkikh A.: Technologies of reconstruction and procedural generation of three-dimensional content. Journal of Physics: Conference Series, 2019, 1399(3), 033018.
- [13] Ge Yu, Tianyu Liu, Ming Liu, Lili Guo: Estimation of Point Cloud Object Pose using Particle Swarm Optimization. Conference Paper April 2018 DOI: 10.1145/3220511.3220512.
- [14] Piatti E. J., Lerma J. L.: A virtual simulator for photogrammetry. (DOI: 10.1111/phor.12001) 2013.
- [15] T. Becker, M. Özkul, U.: Stilla Simulation of close-range photogrammetric systems for industrial surface inspection. PIA11 - Photogrammetric Image Analysis --- Munich, Germany, October 5-7, 2011.
- [16] Gajic D.B., Mihic S., Dragan D., Petrovic V. and Anisic Z.: Simulation of photogrammetry-based 3D data acquisition, 2019.
- [17] Mezhenin, A., Izvozchikova, V., Shardakov V.: Reconstruction of Spatial Environment in Three-Dimensional Scenes. Advances in Intelligent Systems and Computing, 2020, 1127 AISC, c. 47-55.
- [18] Mezhenin, A., Izvozchikova, V., Ivanova, V.: Use of point clouds for video surveillance system cover zone imitation. CEUR Workshop Proceedings, 2019. Vol-2344.
- [19] Krakov D., Feitelson D.: Comparing Performance Heatmaps. DOI:10.1007/978-3-662-43779-7\_3 Corpus ID: 7126407.
- [20] CloudCompare - Open Source project. URL: <https://www.danielgm.net/cc/>.
- [21] Yutaka Ohtake, Alexander Belyaev, Marc Alexa, Greg Turk, Hans-Peter Seidel Multi-level Partition of Unity Implicits SIGGRAPH '05: ACM SIGGRAPH 2005 Courses July <https://doi.org/10.1145/1198555.1198649>.