ISSN 1064-2269, Journal of Communications Technology and Electronics, 2023, Vol. 68, No. 12, pp. 1499–1505. © The Author(s), 2023. This article is an open access publication, corrected publication 2023.
Russian Text © The Author(s), 2023, published in Informatsionnye Protsessy, 2023, Vol. 23, No. 2.1, pp. 289–298.

THEORY AND METHODS OF INFORMATION PROCESSING

Global Refinement Algorithm for 3D Scene Reconstruction from a Sequence of Point Clouds

A. Yu. Makovetskii[a], *, V. I. Kober[b], S. M. Voronin[a], A. V. Voronin[a], V. N. Karnaukhov[b], and M. G. Mozerov[b]

[a] *Chelyabinsk State University, Chelyabinsk, 454001 Russia*

[b] *Kharkevich Institute for Information Transmission Problems, Russian Academy of Sciences, Moscow, 127051 Russia*

**e-mail: artemmac@csu.ru*

Received June 13, 2023; revised June 13, 2023; accepted June 29, 2023

Abstract—Point cloud registration is a central problem in many computer vision problems. However, ensuring global consistency of the results of pairwise registration of point clouds is still a challenge when there are multiple clouds because different scans should be converted to a common coordinate system. This paper describes a global refinement algorithm that first estimates rotations and then estimates parallel translations. For global refinement of rotations, a closed-form algorithm based on matrices is used. For global refinement of parallel translations, a closed-form algorithm is also used. The proposed algorithm is compared with other global refinement algorithms.

Keywords: surface reconstruction, point cloud registration, iterative closest point (ICP) algorithm, variational functional, orthogonal transformation

DOI: 10.1134/S1064226923120124

INTRODUCTION

For effectively solving problems that are performed by mobile robots, it is necessary to construct a 3D model (map) of the space surrounding the robot. An exact map allows the mobile robots to operate under complex conditions by use of only an onboard sensor. Creation of maps of the surrounding medium is called the problem of simultaneous localization and mapping (SLAM). The SLAM problem based on the use of graphs was proposed by Lu and Milios in 1997 [1]. The known approaches to registration of several point clouds consist of the pairwise registration stage and the global refinement stage. Pairwise registration includes attribute matching between pairs of point clouds and minimization of the sum of residuals over all such correspondences to estimate transformation parameters that establish the relative mutual arrangement for each pair of point clouds in the common coordinate system. Pairwise registration involves standard methods of point cloud alignment. The problem of point cloud registration in the three-dimensional space is a fundamental problem of computational geometry and computer vision.

In most cases, global refinement algorithms first find parameters of pairwise transformations by use of [2–14] and then uniformly redistribute errors using graph-based optimization [1, 15]. The graph-based SLAM problem involves the scan graph in which each scanning corresponds to a vertex and each edge corresponds to the spatial connection between pairs of nodes. Globally consistent registration of several point clouds by graph optimization was described in [16]. Further, for the uniform distribution of the error, least squares optimization is used [1]. In [17], the branch-and-bound strategy is used for the global solution of the objective function. The approach proposed in [18] uses surfaces and Bayesian filters for point cloud alignment. The main disadvantage of this method is its high computational cost. Other approaches to global refinement are based on general graph optimization [19], bundle adjustment [20], low-rank sparse decomposition [21], and kernel-based energy function [22].

In [23], the algorithm of global refinement of transformations for point clouds obtained by scanning of the urban environment was described. The algorithm described in [23] first performs global refinement for rotations by use of quaternions and then implements global refinement of parallel translations using the specificity of the urban environment. In the proposed work, for comparisons in computer simulation we use the global refinement algorithm for rotations by use of quaternions as it was presented in [23].

The global refinement algorithm described in the proposed paper first estimates rotations and then estimates parallel translations. For global refinement of rotations, the closed-form algorithm is used by means of matrices. For global refinement of parallel translations, the closed-form algorithm is used.

This paper is organized as follows. Section 1 presents the statement of the problem and describes algorithms of its solution. Section 2 presents results of

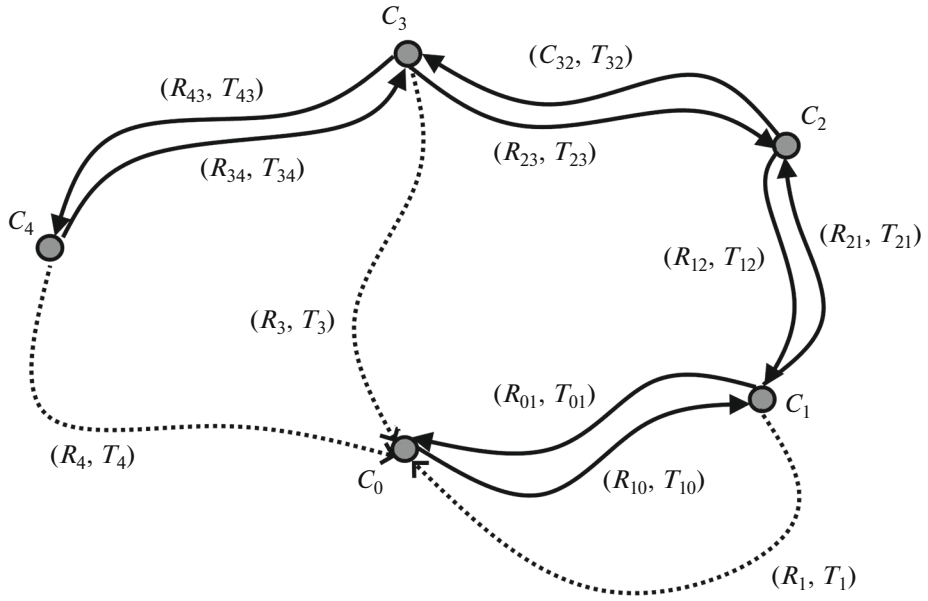

Fig. 1. Graph describing transformations between point clouds.

computer simulation. Section 3 contains the conclusions.

1. GLOBAL REFINEMENT OF RESULTS OF PAIRWISE POINT CLOUD REGISTRATION

Let $C_0, C_1, ..., C_s$ be the initial set of point clouds and (R_{ij}, T_{ij}), $i,j = 0, 1, ..., s$ be results of pairwise cloud registration, where C_j is the reference cloud, C_i is the objective cloud, $R_{ij} \in SO(3)$ is the rotation matrix, and $T_{ij} \in R^3$ is the parallel translation vector.

Let (R_i, T_i), $i = 0, 1, ..., s$, denote transformation mapping of cloud C_i to the coordinate system of cloud C_0. Interpreting point clouds and transformations as vertices and edges, respectively, we obtain a graph, an example of which is shown in Fig. 1. Global refinement of pairwise transformations is based on commutativity of cycles contained in the graph.

1.1. Global Refinement of Rotations

The condition of commutativity of cycles with respect to rotations means that the following conditions are satisfied:

$$R_i = R_j R_{ji}, \tag{1}$$

where $i, j = 0, 1, ..., s$. Let us associate system of equations (1) with functional

$$J'(R) = \sum_{i=0}^{s}\sum_{j=0}^{s}\left\|R_i - R_j R_{ji}\right\|^2, \tag{2}$$

where

$$R = (R_0, ..., R_s).$$

Since the graph can contain not only edges, we replace functional $J'(R)$ by functional $J(R)$:

$$J(R) = \sum_{i=0}^{s}\sum_{j=0}^{s} w_{ji}\left\|R_i - R_j R_{ji}\right\|^2, \tag{3}$$

where

$$w_{ij} = \begin{cases} 1, \text{ edge } (j,i) \text{ is contained in the graph} \\ 0, \text{ edge } (j,i) \text{ is not contained in the graph.} \end{cases}$$

By the solution of system (1) we mean the solution of the following variational problem:

$$R_* = \arg\min_R J(R), \tag{4}$$

under condition $R_i \in SO(3)$, $i = 0, ..., s$. Let $J(R_k)$, $k = 0, 1, ..., s$ denote a functional containing all summands with occurrences of variable R_k in $J(R)$. We represent $J(R_k)$, $k = 0, 1, ..., s$, as a sum of functionals $J_1(R_k)$ and $J_2(R_k)$:

$$J(R_k) = J_1(R_k) + J_2(R_k)$$
$$= \sum_{j=0}^{s} w_{jk}\left\|R_k - R_j R_{jk}\right\|^2 + \sum_{j=0}^{s} w_{kj}\left\|R_j - R_k R_{kj}\right\|^2. \tag{5}$$

Gradients $\nabla J_1(R_k)$ and $\nabla J_2(R_k)$ are specified by formulas:

$$\nabla J_1(R_k) = 2\sum_{j=0}^{s} w_{jk}(R_k - R_j R_{jk}), \tag{6}$$

$$\nabla J_2(R_k) = 2\sum_{j=0}^{s} w_{kj}(R_k R_{kj} R_{kj}^t - R_j R_{kj}^t). \tag{7}$$

Gradient $\nabla J(R_k)$ takes form:

$$\begin{aligned}\nabla J(R_k) &= \nabla J_1(R_k) + \nabla J_2(R_k)\\ &= 2R_k\sum_{j=0}^{s}(w_{jk}I + w_{kj}R_{kj}R_{kj}^t)\\ &- 2\sum_{j=0}^{s}R_j(w_{jk}R_{jk} + w_{kj}R_{kj}^t).\end{aligned} \tag{8}$$

Taking into account that $R_0 = I$, we introduce the following notation:

$$A_{jk} = \begin{cases}-(w_{jk}R_{jk} + w_{kj}R_{kj}^t), & j \neq k\\ \sum\limits_{j=0, j\neq k}^{s}(w_{jk}I + w_{kj}I), & j = k\end{cases}, \tag{9}$$

$$B_k = w_{0k}R_{0k} + w_{k0}R_{k0}^t. \tag{10}$$

Equality $\nabla J(R_k) = 0$ takes form:

$$\sum_{i=1}^{s}A_{ik}R_i = B_k, \tag{11}$$

where $k = 1, ..., s$.

Vanishing of the gradient yields the following linear system of matrix equations:

$$\begin{pmatrix}A_{11} & A_{21} & \ldots & A_{s1}\\ A_{12} & A_{22} & \ldots & A_{s2}\\ \ldots & \ldots & \ldots & \ldots\\ A_{1s} & A_{2s} & \ldots & A_{ss}\end{pmatrix}\begin{pmatrix}R_1\\ R_2\\ \ldots\\ R_s\end{pmatrix} = \begin{pmatrix}B_1\\ B_2\\ \ldots\\ B_s\end{pmatrix}. \tag{12}$$

Let us rewrite system of equations (12) numerically and calculate the affine solution of variational problem (4). We find projections R_{1*}, ..., R_{s*} of obtained matrices R_1, ..., R_s on $SO(3)$:

$$R_{k*} = \begin{cases}U_kV_k^t, \text{ if } \det(U_k)\det(V_k) = 1\\ U_k\mathrm{diag}(1,1,-1)V_k^t, \text{ if } \det(U_k)\det(V_k) = -1\end{cases}, \tag{13}$$

where U_k and V_k are elements of the SVD-representation of matrix R_k.

1.2. Global Refinement of Parallel Translations

The condition of commutativity for cycles of the graph shown in Fig. 1 defines the following system of equations:

$$T_i = R_jT_{ji} + T_j. \tag{14}$$

Let us associate system of equations (14) functional $J(T)$:

$$J(T) = \sum_{i=0}^{s}\sum_{j=0}^{s}w_{ji}\left\|T_i - T_j - R_jT_{ji}\right\|^2, \tag{15}$$

where

$$w_{ij} = \begin{cases}1, \text{ edge } (j,i) \text{ is contained in the graph}\\ 0, \text{ edge } (j,i) \text{ is not contained in the graph}\end{cases}.$$

By the solution of system (14) we mean the solution of the following variational problem:

$$T_* = \arg\min_T J(T), \tag{16}$$

where $T = (T_0, T_1, ..., T_s)$. The gradient $\nabla J(T)$ with respect to T_k, $k = 0, 1, ..., s$ is calculated as follows:

$$\begin{aligned}\nabla J(T) = 2\sum_{i=0}^{s}(w_{ik} + w_{ki})T_k - w_{ik}R_iT_{ik}\\ + w_{ki}R_kT_{ki} - (w_{ik} + w_{ki})T_i.\end{aligned} \tag{17}$$

Vanishing of the gradient with respect to T_k, $k = 0, 1, ..., s$ yields the following equation:

$$\begin{aligned}\sum_{i=0, i\neq k}^{s}-(w_{ik} + w_{ki})T_i + \sum_{i=0, i\neq k}^{s}(w_{ik} + w_{ki})T_k\\ = -\sum_{i=0}^{s}-w_{ik}R_iT_{ik} + w_{ki}R_kT_{ki}.\end{aligned} \tag{18}$$

Let B_k, $k = 0, 1, ..., s$, denote the following expression:

$$B_k = -\sum_{i=0}^{s}-w_{ik}R_iT_{ik} + w_{ki}R_kT_{ki}. \tag{19}$$

Variational problem (16) is reduced to solving a system of linear equations in vectors:

$$\begin{pmatrix}-(w_{10} + w_{01}) & -(w_{10} + w_{01}) & \ldots & -(w_{s0} + w_{0s})\\ \sum\limits_{i=0, i\neq 1}^{s}(w_{i1} + w_{1i}) & -(w_{21} + w_{12}) & \ldots & -(w_{s1} + w_{1s})\\ -(w_{12} + w_{21}) & \sum\limits_{i=0, i\neq 2}^{s}(w_{i2} + w_{2i}) & \ldots & -(w_{s2} + w_{2s})\\ \ldots & \ldots & \ldots & \ldots\\ -(w_{1s} + w_{s1}) & -(w_{2s} + w_{s2}) & \ldots & \sum\limits_{i=0, i\neq s}^{s}(w_{is} + w_{si})\end{pmatrix}\begin{pmatrix}T_1\\ T_2\\ \ldots\\ T_s\end{pmatrix} = \begin{pmatrix}B_0\\ B_1\\ B_2\\ \ldots\\ B_s\end{pmatrix}. \tag{20}$$

Table 1. Accuracy of calculation of rotations by the GR_ICP, GR, and GR_Q algorithms

	GR_ICP	GR	GR_Q
last_R	0.123889	0.0659746	1.36848
avg_R	0.101394	0.0735572	1.062
max_R	0.138166	0.0888956	1.36848

Table 2. Accuracy of calculation of parallel translations by the GR_ICP, GR, and GR_Q algorithms

	GR_ICP	GR	GR_Q
last_T	481.247	35.8375	602.814
avg_T	136.175	18.4498	218.057
max_T	481.247	35.8375	602.814

Let M denote the matrix in Eq. (20). Then, solving the variational problem is reduced to solving three systems of numerical equations:

$$M\begin{pmatrix} T_{1i} \\ T_{2i} \\ \dots \\ T_{si} \end{pmatrix} = \begin{pmatrix} B_{0i} \\ B_{1i} \\ B_{2i} \\ \dots \\ B_{si} \end{pmatrix}, \quad (21)$$

where $i = 1, 2, 3$ is the number of the vector component.

2. COMPUTER SIMULATION

We denote the global refinement algorithm proposed in this paper as GR. Let us describe other algorithms under consideration.

For cloud C_k, $k = 1, ..., s$, we consider transformation $M_{k(k-1)}$ equal to the result of projection of matrix $1/2(R_{(k-1)k} + R_{k(k-1)})$ on $SO(3)$. Let T_k, $k = 1, ..., s$, denote the parallel translation vector equal to vector $1/2(T_{(k-1)k} + T_{k(k-1)})$. The transformation $((M_1M_2 \dots M_k), (T_1 + T_2 + \dots + T_k))$ maps cloud C_k to the coordinate system of cloud C_0. We denote this global refinement algorithm as R_ICP.

In [23], the global refinement algorithm was described individually for rotations and parallel translations. The algorithm for rotations is based on using quaternions. Let GR_Q denote the global refinement algorithm that uses the rotation refinement algorithm described in [23] and the algorithm for parallel translation described in this paper.

Table 3. Accuracy of calculation of rotations by the GR_ICP, GR, and GR_Q algorithms

	GR_ICP	GR	GR_Q
last_R	0.140062	0.140093	0.0118888
avg_R	0.0661191	0.0661755	0.0092607
max_R	0.140062	0.140093	0.0242108

The computer experiments were carried out with point clouds from the San Francisco Apollo-South-Bay Dataset [24]. Each cloud in the database contains approximately 100000 points. We use cloud subsampling to approximately 10000 points per cloud. The clouds in the dataset were obtained using a lidar mounted on a vehicle. The vehicle moved and the sensor scanned the ambient medium with a certain frequency. The obtained data set consists of a sequence of point clouds. In our experiments, the point clouds are taken from the dataset with a step of 4, i.e., for example, point clouds nos. 1, 5, 9, ... are considered.

The database contains information about transformation M_k mapping each cloud C_k to a certain global coordinate system. Matrix M_k has dimensions of 4 by 4 and specifies a rigid transformation in homogeneous coordinates. The transformation mapping cloud C_i to the coordinate system of cloud C_J is specified by matrix $M_{ji\_true} = (M_j)^{-1}M_i$.

The experiments are organized as follows. Point cloud number k is fixed in the database. The point clouds with numbers k, $k + 4$, $k + 8$, $...k + 4 \times 99$ are considered. For all successive pairs of clouds $(k + 4i, k + 4(i + 1))$, $i = 0, ..., 98$, we find transformation $(R_{k+4i,\ k+4(i+1)},\ T_{k+4i,\ k+4(i+1)})$ mapping cloud $C_{k+4(i+1)}$ to the coordinate system of cloud C_{k+4i} and transformation $(R_{k+4(i+1),\ k+4i},\ T_{k+4(i+1),\ k+4i})$ mapping cloud C_{k+4i} to the coordinate system of cloud $C_{k+4(i+1)}$ using the point-to-point ICP algorithm. Note that before the use of the ICP algorithm we apply to the reference cloud the coarse alignment algorithm, which means applying to this cloud the transformation obtained for the previous pair of clouds. The transforms relating clouds, the numbers of which differ by more than 4, are calculated using the superposition of intermediate transforms with respect to R and T, respectively.

In this paper, the following quality parameters of global refining algorithms are used. Parameter last_R = $\|R_{(\text{first, last})\_\text{true}} - R_{(\text{first, last})\_\text{est}}\|_{L_2}$, where $R_{(\text{first, last})\_\text{true}}$ and $R_{(\text{first, last})\_\text{est}}$ are the true and estimated transforms mapping the last considered cloud to the coordinate system of the first cloud, shows the global error of 3D scene reconstruction. Parameter last_T is defined similarly. Parameters avg_R and avg_T show the average errors with respect to R and T, respectively. Parameters max_R and max_T show the maximum errors. Figure 2 and Tables 1 and 2 show the operation accuracy of the GR_ICP, GR, and GR_Q algorithms for a series of clouds with the origin in cloud no. 1.

Figure 3 and Tables 3 and 4 show the operation accuracy of the GR_ICP, GR, and GR_Q algorithms for a series of clouds with the origin in cloud no. 501.

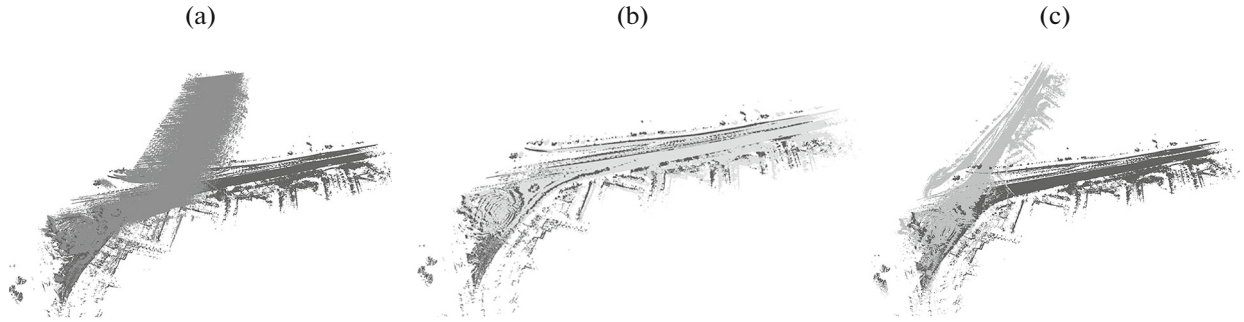

Fig. 2. Result of 3D scene formation by different algorithms: (a) GR_ICP (dark gray color); (b) GR (light gray color); and (c) GR_Q (gray color). The correctly formed 3D scene is marked by black color.

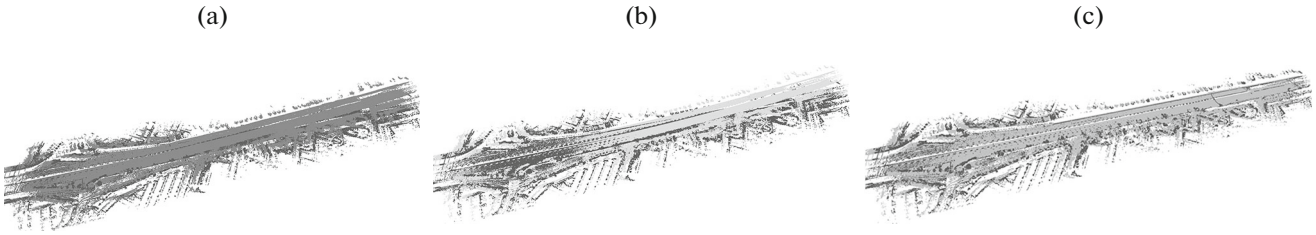

Fig. 3. Result of 3D scene formation by different algorithms: (a) GR_ICP (dark gray color); (b) GR (light gray color); and (c) GR_Q (gray color). The correctly formed 3D scene is marked by black color.

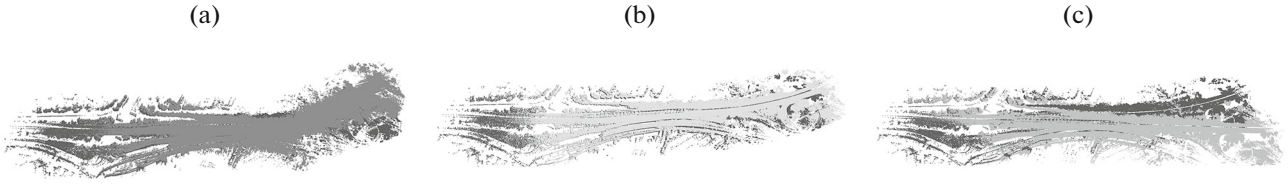

Fig. 4. Result of 3D scene formation by different algorithms: (a) GR_ICP (dark gray color); (b) GR (light gray color); and (c) GR_Q (gray color). The correctly formed 3D scene is marked by black color.

Figure 4 and Tables 5 and 6 show the operation accuracy of the GR_ICP, GR, and GR_Q algorithms for a series of clouds with the origin in cloud no. 1001. Figure 5 and Tables 7 and 8 show the operation accuracy of the GR_ICP, GR, and GR_Q algorithms for a series of clouds with the origin in cloud no. 1501.

3. CONCLUSIONS

In this paper, we use the global refining algorithm GR for constructing a 3D scene from a set of point clouds. The algorithm is compared with other possible methods of solving the problem of global refining of transformations. The accuracy of the algorithms is

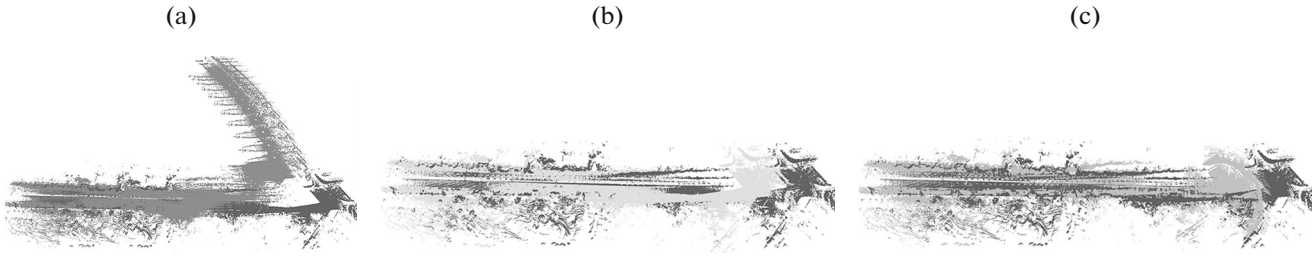

Fig. 5. Result of 3D scene formation by different algorithms: (a) GR_ICP (dark gray color); (b) GR (light gray color); and (c) GR_Q (gray color). The correctly formed 3D scene is marked by black color.

Table 4. Accuracy of calculation of parallel translations by the GR_ICP, GR, and GR_Q algorithms

	GR_ICP	GR	GR_Q
last_T	55.7511	54.1935	17.2386
avg_T	27.6695	25.5408	16.1963
max_T	55.9709	54.1935	17.2544

Table 5. Accuracy of calculation of rotations by the GR_ICP, GR, and GR_Q algorithms

	GR_ICP	GR	GR_Q
last_R	0.104861	0.0933781	0.443587
avg_R	0.0573034	0.0510654	0.195734
max_R	0.117359	0.105829	0.443587

Table 6. Accuracy of calculation of parallel translations by the GR_ICP, GR, and GR_Q algorithms

	GR_ICP	GR	GR_Q
last_T	92.9949	36.5377	136.19
avg_T	38.0313	12.4912	47.7568
max_T	92.9949	36.5377	136.19

Table 7. Accuracy of calculation of rotations by the GR_ICP, GR, and GR_Q algorithms

	GR_ICP	GR	GR_Q
last_R	0.108476	0.0784722	1.13145
avg_R	0.057009	0.0474452	0.164958
max_R	0.108679	0.0835886	1.13145

Table 8. Accuracy of calculation of parallel translations by the GR_ICP, GR, and GR_Q algorithms

	GR_ICP	GR	GR_Q
last_T	579.766	131.92	126.563
avg_T	164.922	124.399	123.483
max_T	579.766	136.542	136.57

estimated using quality criteria that correspond to visual perception of the result accuracy. Computer simulation demonstrates efficiency of the proposed algorithm.

FUNDING

This work was supported in part by the Russian Science Foundation, project no. 21-11-00095.

CONFLICT OF INTEREST

The authors of this work declare that they have no conflicts of interest.

OPEN ACCESS

REFERENCES

1. F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," Autonomous Robots **4**, 333–349 (1997).
2. P. Besl and N. McKay, "A method for registration of 3-D shapes," IEEE Trans. Pattern Anal. and Machine Intell. **2**, 239–256 (1992).
3. Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," Image and Vision Comput. **10**, 145–155 (1992).
4. A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," Robot. Sci. Syst. **5**, 161–168 (2010).
5. J. Serafin and G. Grisetti, "Using extended measurements and scene merging for efficient and robust point cloud registration," Robot. Auton. Syst. **92**, 91–106 (2017).
6. A. Makovetskii, S. Voronin, V. Kober, and A. Voronin, "A regularized point cloud registration approach for orthogonal transformations," J. Global Optimiz (2020).
7. A. Makovetskii, S. Voronin, V. Kober, and D. Tihonkih, "Affine registration of point clouds based on point-to-plane approach," Procedia Eng. **201**, 322–330 (2017).
8. B. Horn, "Closed-form solution of absolute orientation using unit quaternions," J. Opt. Soc. America, Ser. A **4**, 629–642 (1987).
9. B. Horn, H. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using orthonormal matrices," J. Opt. Soc. America. Ser. A **5**, 1127–1135 (1988).
10. S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," IEEETPAMI **13** (4), 376–380 (1991).
11. A. Makovetskii, S. Voronin, V. Kober, and A. Voronin, "A point-to-plane registration algorithm for orthogonal transformations," Proc. SPIE, 10752R (2018).

12. A. Makovetskii, S. Voronin, V. Kober, and A. Voronin, "A non-iterative method for approximation of the exact solution to the point-to-plane variational problem for orthogonal transformations," Math. Methods Appl. Sci. **41** (18), 9218 –9230 (2018).
13. A. Makovetskii, S. Voronin, V. Kober, and A. Voronin, "Point cloud registration based on multiparameter functional," Mathematics **9**, 2589 (2021).
14. A. Makovetskii, S. Voronin, V. Kober, and A. Voronin, "Coarse point cloud registration based on variational functional," Mathematics **11**, 35 (2023).
15. D. Bornnmann, J. Elseberg, K. Lingmann, A. Nuechter, and J. Hertzberg, "Globally Consistent, 3D mapping with scan matching," Robot. Auton. Syst. **56**, 130–142 (2008).
16. P. W. Theiler, J. D. Wegner, and K. Schindler, "Globally consistent registration of terrestrial laser scans via graph optimization," ISPRS J. Photogram. Remote Sens. **109**, 126–138 (2015).
17. J. Yang, H. Li, and Y. Jia, "Go-ICP: Solving 3D registration efficiently and globally optimally," in *Proc. 2013 IEEE Int. Conf. on Comput. Vision, Sydney, NSW, Australia, Dec. 1–8, 2013* (IEEE, New York, 2013).
18. D. Huber and M. Hebert, "Fully automatic registration of multiple 3D data sets," Image Vis. Comput. **21**, 637–650 (2003).
19. R. Kuemmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "A general framework for graph optimization," in *Proc. 2011 IEEE Int. Conf. Robotics and Automation, Shanghai, China, May 9–13, 2011* (IEEE, New York, 2011), pp. 3607–3613.
20. M. A. Lourakis and A. Argyros, "SBA: A software package for generic sparse bundle adjustment," ACM Trans. Math. Softw. **36**, 1–30 (2009).
21. S. Wang, H. Y. Sun, H. C. Guo, L. Du, and T. J. Liu, "Multi-view laser point cloud global registration for a single object," Sensors **18**, 3729 (2018).
22. S. McDonagh and F. Robert, "Simultaneous registration of multi-view range images with adaptive kernel density estimation," in *Proc. IMA, 14th Mathematics of Surfaces, Birmingham, AL, USA, Sept., 11–13, 2013.*
23. Nadisson Luis Pavan, Daniel Rodrigues dos Santos, and Khoshelham Kourosh, "Global registration of terrestrial laser scanner point clouds using plane-to-plane correspondences," Remote Sens. **12**, 1127 (2020).
24. W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, "L3-Net: Towards learning based LiDAR localization for autonomous driving," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition, Long Beach, USA, Iune 15–20, 2019* (IEEE, New York, 2019), pp. 6389–6398.

Translated by A. Nikol'skii