



# A regularized point cloud registration approach for orthogonal transformations

Artyom Makovetskii<sup>1</sup> · Sergei Voronin<sup>1</sup> · Vitaly Kober<sup>1,2,3</sup> · Aleksei Voronin<sup>1</sup>

Received: 14 January 2020 / Accepted: 13 July 2020 / Published online: 22 July 2020  
© Springer Science+Business Media, LLC, part of Springer Nature 2020

## Abstract

An important part of the well-known iterative closest point algorithm (ICP) is the variational problem. Several variants of the variational problem are known, such as point-to-point, point-to-plane, generalized ICP, and normal ICP (NICP). This paper proposes a closed-form exact solution for orthogonal registration of point clouds based on the generalized point-to-point ICP algorithm. We use points and normal vectors to align 3D point clouds, while the common point-to-point approach uses only the coordinates of points. The paper also presents a closed-form approximate solution to the variational problem of the NICP. In addition, the paper introduces a regularization approach and proposes reliable algorithms for solving variational problems using closed-form solutions. The performance of the algorithms is compared with that of common algorithms for solving variational problems of the ICP algorithm. The proposed paper is significantly extended version of Makovetskii et al. (CCIS 1090, 217–231, 2019).

**Keywords** Variational functionals · Global optimization · Exact solution · Closed-form solution · Iterative closest points (ICP) · Normal ICP (NICP) · Affine transformations · Orthogonal transformations · Surface reconstruction · Computer geometry

## 1 Introduction

Creating a 3D spatial environment for a robot or sensor is based on algorithms for registering point clouds. Point cloud registration methods have been the subject of research since the early 1990s, and a wide range of options have been proposed. Aligning two point clouds means finding either an orthogonal or affine transformation in  $R^3$  that maximizes consistent

---

The work was supported by the RFBR (Grant 18-07-00963).

---

✉ Artyom Makovetskii  
artemmac@csu.ru

<sup>1</sup> Chelyabinsk State University, Chelyabinsk, Russian Federation 454001

<sup>2</sup> CICESE, 22860 Ensenada, B.C., Mexico

<sup>3</sup> Institute for Information Transmission Problems, Russian Academy of Sciences, Moscow, Russian Federation 127051

overlap between the two clouds. The iterative closest points algorithm (ICP) is the most common method for aligning point clouds based on exclusively geometric characteristics. The algorithm is widely used to record data obtained with 3D scanners. The ICP algorithm, originally proposed by Besl and McKay [2], Chen and Medioni [3], consists of the following iteratively applied basic steps:

- (1) search for correspondence between the points of two clouds;
- (2) minimization of the error metric (variational problem of the ICP algorithm).

The two steps of the ICP algorithm alternate among themselves, that is, the estimation of the geometric transformation based on the fixed correspondence (step 2) and updating the correspondences to their closest matches (step 1). The key point of the ICP algorithm [4] is the search for either an orthogonal or affine transformation, which is the best with respect to a metric for combining two point clouds with a given correspondence between points.

The variational problem of the ICP algorithm contains the following three basic components: functional to be minimized; class of geometric transformations; functional minimization method. The most common types of functionals are presented in different variants of the ICP; that is, point-to-point [2], point-to-plane [3], generalized ICP [5], and normal ICP (NICP) [6]. Geometric transformations can belong to the groups of  $GL(3)$ ,  $O(3)$ ,  $SO(3)$  (affine transformations, orthogonal transformations, orthogonal transformations with positive determinants, respectively) extended by translations. A minimization method can be either iterative or closed (closed-form solution). A closed-form solution can be an exact solution to a variational problem or its approximation. Among all iterative methods, Levenberg-Marquardt, Gauss-Newton, conjugate gradients are most often used. For example, the variational problem [2] contains the point-to-point functional, the transformation class of  $SO(3)$ , and the Gauss – Newton minimization method. Note that the solution to a variational problem in the class of orthogonal transformations is mathematically more complicated than in the class of affine transformations, since in the former case it is necessary to deal with the manifold  $O(3)$  (or  $SO(3)$ ) in  $R^9$ .

Many different variants of the variational problem have been proposed. In [7] is described closed form solution for point-to-point affine problem. Closed form solutions of the affine point-to-plane problem are described in [8,9]. Closed-form solutions to the point-to-point problem in the class of orthogonal transformations were obtained by Horn [10,11]. In [10], the solution is based on quaternions and belongs to  $SO(3)$ . In [11], the transformation matrix belongs to  $O(3)$  and may have negative determinant, and therefore, the ICP algorithm may not converge. This problem was solved by modifying the Horn's algorithm for the class of  $SO(3)$  in [12]. More robust ICP algorithms with other functionals called the generalized ICP [5] and NICP [6] were also proposed. The corresponding variational problems are solved by iterative conjugate gradients and Gauss-Newton methods.

Typically, the ICP algorithm monotonically reduces the functional values, but owing to the problem non-convexity, the algorithm often stops at suboptimal local minima. The accuracy of the solution depends on the quality of correspondence between source and target point clouds. Suppose that the target cloud is obtained from the source cloud by a true orthogonal transformation, and the ideal correspondence between the points of the source and target clouds is known. In this case, any variational problem finds the true orthogonal transformation (with obvious conditions for iterative methods). For example, in this case the simplest point-to-point variational problem (for  $GL(3)$  class) [7] finds the true (even orthogonal) transformation. Note that in real applications, the correspondence between point clouds is far from ideal. If the source and target clouds are located far from each other, then a common algorithm of searching for correspondence between clouds matches all points of the source

cloud with a small subset of the target cloud. In this case the affine variational problem finds a transformation that strongly distorts the source cloud. Also the bad correspondence significantly reduces the probability to obtain a good answer for orthogonal variants of variational problems. Thus, the probability of obtaining an acceptable transformation as a result of the ICP algorithm with initial poor correspondence is the comparative criterion for different types of variational problems.

Experimental comparative analysis of the NICIP algorithm, generalized ICP, NDT [13], and KinFu [14] is presented [6]. The analysis shows that the NICIP yields better results and higher robustness compared to other state-of-the-art methods.

In [1] there is proposed the functional of generalized point-to-point ICP variational problem and a method to solve it. The proposed paper is significantly extended version of [1].

The proposed paper consists of the two main parts. The first part describes a variational problem in which information on the coordinates of points and normal vectors of point cloud is used. We offer a closed-form exact solution to this variational problem. Since the solution of the variational problem [1] using the method from [11] can have a rotation matrix with negative determinant, in this paper we propose a closed-form exact solution to the variational problem in the class of  $SO(3)$  using the method from [12]. This variational problem with regularization is denoted as  $\lambda\_MH-RICP$ .

Note that the  $\lambda\_MH-RICP$  functional without regularization term coincides with the NICIP functional [6] if all point information matrices are identity matrices, and all normal information matrices are  $diag(\lambda)$ . To solve the variational problem, the authors [6] use the iterative Gauss-Newton method based on quaternion parametrization of  $SO(3)$ , while in this paper we derive a closed-form exact solution.

In the second part of the paper, we deal with the variational problem with a functional that coincides with the NICIP functional, but for arbitrary information matrices. To solve the variational problem, a three-stage method is proposed. First, we find a solution to the variational problem in the class of  $GL(3)$ . Second, we project the obtained onto the submanifold  $SO(3)$  in  $R^9$ . Third, the translation vector is computed. These three steps offer a closed-form solution to the variational problem, which approximates the exact solution. A similar approximation method is used [15] for point-to-plane ICP. This variational problem with regularization is denoted as  $\Omega\_MH-RICP$ . The  $\Omega\_MH-RICP$  functional (without regularization term) coincides with NICIP functional.

Here we emphasize that the variational problems of  $\Omega\_MH-RICP$  (for the zero value of the regularization parameter  $\alpha$ ) and NICIP in terms of the three basic components have the same first and second elements and different third elements.

The aim of the different types of variational problems implementation is increasing a probability to find a suitable transformation between point clouds with poor initial correspondence. In the proposed paper we compare the  $\Omega\_MH-RICP$  and NICIP variational problems, since NICIP variational problem offers better results than other known methods [6].

An analysis of the ICP algorithm work allows us to formulate the regularized variant of the ICP algorithm (RICP). We add an additional regularization term with the parameter  $\alpha$  to the functional. The use of the regularization term increases the probability of obtaining a good result of the ICP algorithm work with poor correspondence in the first steps of the ICP algorithm. The experimental results show that convergence of the  $\lambda\_MH-RICP$  and  $\Omega\_MH-RICP$  methods is better than the convergence of the  $\lambda\_MH-ICP$  and  $\Omega\_MH-ICP$  methods without regularization respectively (in case of poor correspondence). In such a way the proposed variant of the ICP variational problem improves the quality of the ICP algorithm work to solve the global optimization problem.

Note that the proposed approach can be applied to various ICP variational functionals.

With the help of extensive computer simulation, we show that the proposed algorithms yield good convergence to a suitable transformation, even when the correspondence of points is estimated incorrectly at the first iterations.

The paper is organized as follows. In Sect. 2, a closed-form exact solution to the generalized point-to-point variational problem in the class of  $SO(3)$  is presented. In Sect. 3, we propose a closed-form approximation of the exact solution to the variation problem with NICP functional. In Sect. 4, computer simulation results are presented and discussed. Section 5 summarizes our conclusions.

## 2 Generalized point-to-point approach

Let  $P = \{p^1, \dots, p^s\}$  be a source point cloud, and  $Q = \{q^1, \dots, q^s\}$  be a target point cloud in  $R^3$ . Suppose that the relation between points of  $P$  and  $Q$  is given in such a manner that for each point  $p^i$  exists a corresponding point  $q^i$ . Denote the surfaces constructed from the clouds  $P$  and  $Q$  by  $S(P)$  and  $S(Q)$  respectively; let  $T_P(p^i)$  and  $T_Q(q^i)$  be the tangent planes of  $S(P)$  and  $S(Q)$  at points  $p^i$  and  $q^i$ , respectively.

The ICP algorithm is defined as a geometrical transformation for rigid objects, mapping  $P$  to  $Q$

$$Rp_i + T, \quad R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}, \quad p^i = (p_1^i \ p_2^i \ p_3^i)^t, \quad q^i = (q_1^i \ q_2^i \ q_3^i)^t, \quad (1)$$

where  $R$  is a rotation (orthogonal) matrix,  $T$  is a translation vector,  $i = 1, \dots, s$ . Denote by  $J_h(R, T)$  the following functional:

$$J_h(R, T) = \sum_{i=1}^s \|Rp^i + T - q^i\|^2. \quad (2)$$

The point-to-point variational problem is formulated as follows: to find

$$\arg \min_{R, T} J_h(R, T). \quad (3)$$

Let  $n_p^i$  and  $n_q^i$  be the normal vectors to planes  $T_P(p^i)$  and  $T_Q(q^i)$ , respectively,  $i = 1, \dots, s$ .

**Remark 1** Suppose that normal vectors are given. The coordinates of the vectors can be provided initially with the point clouds or calculated from the local context [6].

Consider the functional  $J_g(R, T)$  as

$$J_g(R, t) = \sum_{i=1}^s \|Rp^i + T - q^i\|^2 + \lambda \sum_{i=1}^s \|Rn_p^i - n_q^i\|^2, \quad (4)$$

where  $\lambda$  is a parameter. A the generalized point-to-point variational problem is given as follows: to find

$$\arg \min_{R, T} J_g(R, T). \quad (5)$$

Let us define a functional  $J_{rg}(R, T)$  as

$$J_{rg}(R, T) = \alpha \| R - I \|^2 + \sum_{i=1}^s \| Rp^i + T - q^i \|^2 + \lambda \sum_{i=1}^s \| Rn_p^i - n_q^i \|^2, \tag{6}$$

where  $\alpha$  is a regularization parameter,  $I$  is the identity matrix. The corresponding variational problem is looking for

$$arg \min_{R,T} J_{rg}(R, T). \tag{7}$$

**Remark 2** The introduction of the proposed regularization term can be explained by the following way. Since the initial steps of the ICP algorithm often show poor correspondence between point clouds, this can lead to random rotation with large angles of the clouds relative to each other. In this case, the probability of finding good correspondence between the clouds for the next steps of the ICP algorithm is low, and the ICP algorithm often stops at a suboptimal local minimum. Sufficiently large values of the parameter  $\alpha$  and the proposed form of regularization prevent large rotation angles at the first stages of the ICP algorithm. On the other hand, the use of large values of the regularization parameter  $\alpha$  even with a good correspondence between point clouds often lead to poor results, while the processing time of the ICP algorithm after the first few steps increases significantly. Therefore, the parameter value should be reduced after the first steps of the ICP algorithm.

Let us denote versions of the ICP algorithm based on variational problems (5) and (7) as  $\lambda\_MH\text{-ICP}$  and  $\lambda\_MH\text{-RICP}$ , respectively.

### 2.1 Exclusion of translation vector

Let us apply to all points of the cloud  $P$  the following transformation:

$$\begin{cases} (p')_1^i = p_1^i - \frac{1}{s} \sum_{i=1}^s p_1^i \\ (p')_2^i = p_2^i - \frac{1}{s} \sum_{i=1}^s p_2^i \\ (p')_3^i = p_3^i - \frac{1}{s} \sum_{i=1}^s p_3^i \end{cases}, \tag{8}$$

and the corresponding transformation to points of the cloud  $Q$ ,

$$\begin{cases} (q')_1^i = q_1^i - \frac{1}{s} \sum_{i=1}^s q_1^i \\ (q')_2^i = q_2^i - \frac{1}{s} \sum_{i=1}^s q_2^i \\ (q')_3^i = q_3^i - \frac{1}{s} \sum_{i=1}^s q_3^i \end{cases}, \tag{9}$$

where  $i = 1, \dots, s$ . We get new clouds  $P'$  and  $Q'$  from  $P$  and  $Q$  by translations. For clouds  $P'$  and  $Q'$ , the functional (4) takes the form

$$J_g(R) = \sum_{i=1}^s \| R(p')^i - (q')^i \|^2 + \lambda \sum_{i=1}^s \| Rn_p^i - n_q^i \|^2, \tag{10}$$

where  $(p)^i$  and  $(q)^i$  are points of the clouds  $P'$  and  $Q'$ ,  $i = 1, \dots, s$ . The variational problem (5) takes the following form: to find

$$arg \min_R J_g(R). \tag{11}$$

The functional (6) and variational problem (7) can be rewritten respectively as

$$J_{rg}(R) = \alpha \| R - I \|^2 + \sum_{i=1}^s \| R(p')^i - (q')^i \|^2 + \lambda \sum_{i=1}^s \| Rn_p^i - n_q^i \|^2, \tag{12}$$

to find

$$\operatorname{arg\,min}_R J_{rg}(R). \tag{13}$$

**Remark 3** We will use the notation  $p^i$  and  $q^i$  in Sects. 2.2 and 2.3 instead of  $(p')^i$  and  $(q')^i$ , respectively, for simplicity. The original notation  $(p')^i$  and  $(q')^i$  will be used again in Sect. 2.4.

### 2.2 Reduction of the variational problem

Let us reduce the variational problem (11) to a problem that can be solved by the Horn’s method [11]. The functional (10) can be rewritten as

$$\begin{aligned} J_g(R) &= \sum_{i=1}^s \langle Rp^i - q^i, Rp^i - q^i \rangle + \lambda \sum_{i=1}^s \langle Rn_p^i - n_q^i, Rn_p^i - n_q^i \rangle \\ &= \sum_{i=1}^s (\langle Rp^i, Rp^i \rangle - 2 \langle Rp^i, q^i \rangle + \langle q^i, q^i \rangle) \\ &\quad + \lambda \sum_{i=1}^s (\langle Rn_p^i, Rn_p^i \rangle - 2 \langle Rn_p^i, n_q^i \rangle + \langle n_q^i, n_q^i \rangle), \end{aligned} \tag{14}$$

where  $\langle \cdot, \cdot \rangle$  denotes the inner product. Note that expressions  $\langle q^i, q^i \rangle$  and  $\langle n_q^i, n_q^i \rangle$  do not depend on  $R$ . Therefore, these expressions do not affect the variational problem (11). The functional  $J_g(R)$  takes the form

$$\begin{aligned} J_g(R) &= \sum_{i=1}^s (\langle Rp^i, Rp^i \rangle - 2 \langle Rp^i, q^i \rangle) \\ &\quad + \lambda \sum_{i=1}^s (\langle Rn_p^i, Rn_p^i \rangle - 2 \langle Rn_p^i, n_q^i \rangle) + \operatorname{const} \\ &= \sum_{i=1}^s (\langle R^t Rp^i, p^i \rangle - 2 \langle Rp^i, q^i \rangle) \\ &\quad + \lambda \sum_{i=1}^s (\langle R^t Rn_p^i, n_p^i \rangle - 2 \langle Rn_p^i, n_q^i \rangle) + \operatorname{const}. \end{aligned} \tag{15}$$

Since  $R$  is an orthogonal matrix, then  $R^t R = E$ . The inner products  $\langle p^i, p^i \rangle$  and  $\langle n_p^i, n_p^i \rangle$  do not depend on  $R$ . It follows that the functional  $J_g(R)$  takes the form

$$J_g(R) = -2 \left( \sum_{i=1}^s \langle Rp^i, q^i \rangle + \lambda \sum_{i=1}^s \langle Rn_p^i, n_q^i \rangle \right) + \operatorname{const}. \tag{16}$$

The variational problem (11) can be rewritten as follows: to find

$$\operatorname{arg\,max}_R \left( \sum_{i=1}^s \langle Rp^i, q^i \rangle + \lambda \sum_{i=1}^s \langle Rn_p^i, n_q^i \rangle \right). \tag{17}$$

Note that the inner product  $\langle Rp^i, q^i \rangle$  can be expressed by the matrix trace

$$\langle Rp^i, q^i \rangle = \operatorname{tr}(R \cdot (p^i (q^i)^t)). \tag{18}$$

Denote the matrix  $p^i (q^i)^t$  by  $M^i$ . It means that

$$\begin{aligned} \sum_{i=1}^s \langle Rp^i, q^i \rangle &= \sum_{i=1}^s \operatorname{tr}(R \cdot (p^i (q^i)^t)) = \sum_{i=1}^s \operatorname{tr}(R \cdot M^i) \\ &= \operatorname{tr} \left( \sum_{i=1}^s (R \cdot M^i) \right) = \operatorname{tr} \left( R \cdot \sum_{i=1}^s M^i \right). \end{aligned} \tag{19}$$

Let denote the matrix  $D_p$  as

$$D_p = \sum_{i=1}^s M^i. \tag{20}$$

Then we can write

$$\sum_{i=1}^s \langle Rp^i, q^i \rangle = \operatorname{tr}(R \cdot D_p). \tag{21}$$

Note that the following condition holds:

$$\sum_{i=1}^s \langle Rp^i, q^i \rangle = \operatorname{tr}(R \cdot D_p) = \langle R, (D_p)^t \rangle, \tag{22}$$

where  $\langle R, (D_p)^t \rangle$  denotes the sum of the products of the corresponding matrix elements. Denote the matrix  $n_p^i (n_q^i)^t$  by  $M_n^i$ , and let  $D_n$  be the following matrix:

$$D_n = \sum_{i=1}^s M_n^i. \tag{23}$$

In a similar way, we get

$$\sum_{i=1}^s \langle Rn_p^i, n_q^i \rangle = \operatorname{tr}(R \cdot D_n) = \langle R, (D_n)^t \rangle. \tag{24}$$

The variational problem (17) can be rewritten as follows: to find

$$\begin{aligned} \operatorname{arg\,max}_R \left( \sum_{i=1}^s \langle Rp^i, q^i \rangle + \lambda \sum_{i=1}^s \langle Rn_p^i, n_q^i \rangle \right) \\ = \operatorname{arg\,max}_R (\langle R, (D_p)^t \rangle + \lambda \langle R, (D_n)^t \rangle). \end{aligned} \tag{25}$$

Let  $D$  be the following matrix:

$$D = (D_p)^t + \lambda (D_n)^t. \tag{26}$$

Then the variational problem (25) takes the following form: to find

$$\operatorname{arg\,max}_R \langle R, D \rangle. \tag{27}$$

So, the variational problem (5) is reduced to (27). In a similar manner, the variational problem (7) can be reduced to the following problem: to find

$$\operatorname{arg\,max}_R \langle R, D + \alpha I \rangle. \tag{28}$$

**Remark 4** The solution (27) was obtained [1] using the Horn’s method [11]. The problem (28) can be solved in a similar way. The disadvantage of this approach is that the solution belongs to the class of transformations  $O(3)$ . Next, we obtain a closed-form exact solution to the variational problem in the class of transformations  $SO(3)$  using the method from [12].

### 2.3 Closed-form solution to the variational problem

In [12] is considered the following variational problem: to find

$$\operatorname{arg\,min}_R \| RP - Q \|^2, \tag{29}$$

where  $3 \times s$  matrix  $P$  consists of the vector-columns  $p^i$ ,  $3 \times s$  matrix  $Q$  is composed of the vector-columns  $q^i$ ,  $i = 1, \dots, s$ . The variational problem (29) is equivalent to the standard point-to-point problem after exclusion of the translation vector. The problem (29) has a solution  $USV^t$ , where matrices  $U$  and  $V$  are elements of a singular value decomposition  $UDV^t$  of the matrix  $QP^t$  (suppose that  $D$  is a diagonal matrix with non-increasing diagonal elements), and  $S$  is defined as

$$S = \begin{cases} I, & \text{if } \det(U)\det(V) = 1 \\ \operatorname{diag}(1, 1, \dots, 1, -1), & \text{if } \det(U)\det(V) = -1, \end{cases} \tag{30}$$

To solve the variational problems (3), (5) and (9), the matrix  $QP^t$  is replaced by the matrices  $(D_p)^t$  (see (20)),  $D$  (see (26)) and  $(D + \alpha I)$ , respectively. Solutions to these variational problems are orthogonal matrices with positive determinants.

### 2.4 Return from $P'$ and $Q'$ to $P$ and $Q$

Let matrix  $R_*$  be a solution to the variational problem (27). Then  $R_*$  is also a solution to the variational problem (11)

$$R_* = \operatorname{arg\,min}_R \left( \sum_{i=1}^s \| R(p')^i - (q')^i \|^2 + \lambda \sum_{i=1}^s \| Rn_p^i - n_q^i \|^2 \right). \tag{31}$$

Denote by  $v_p$  and  $v_q$  the following vectors:

$$v_p = \left( \sum_{i=1}^s p_1^i, \sum_{i=1}^s p_2^i, \sum_{i=1}^s p_3^i \right)^t, \quad v_q = \left( \sum_{i=1}^s q_1^i, \sum_{i=1}^s q_2^i, \sum_{i=1}^s q_3^i \right)^t. \tag{32}$$



Rewrite (8) and (9) as

$$(p')^i = p^i - \frac{1}{s} v_p, \tag{33}$$

$$(q')^i = q^i - \frac{1}{s} v_q. \tag{34}$$

The functional in (31) can be rewritten by taking into account (33) and (34) as

$$\begin{aligned} & \sum_{i=1}^s \left\| R(p^i - \frac{1}{s} v_p) - (q^i - \frac{1}{s} v_q) \right\|^2 + \lambda \sum_{i=1}^s \left\| Rn_p^i - n_q^i \right\|^2 \\ &= \sum_{i=1}^s \left\| Rp^i - q^i + \frac{1}{s}(v_q - Rv_p) \right\|^2 + \lambda \sum_{i=1}^s \left\| Rn_p^i - n_q^i \right\|^2. \end{aligned} \tag{35}$$

It can be seen that the matrix  $R_*$  minimizes both the left and right sides of the Eq. (35). It means that the optimal translation vector  $T_*$  is obtained as

$$T_* = \frac{1}{s}(v_q - R_*v_p). \tag{36}$$

The matrix  $R_*$  and the vector  $T_*$  are solutions to the variational problem (5). In the same way, the vector  $T_*$  for the variational problem (7) can be computed.

### 3 Closed-form approximation of the exact solution to the NICP variational problem

Denote by  $J_\Omega$  the following functional:

$$\begin{aligned} J_\Omega(R, T) &= \alpha \left\| R - I \right\|^2 \\ &+ \sum_{i=1}^s (Rp^i + T - q^i)^t \Omega_i^p (Rp^i + T - q^i) \\ &+ \sum_{i=1}^s (Rn_p^i - n_q^i)^t \Omega_i^n (Rn_p^i - n_q^i), \end{aligned} \tag{37}$$

where  $\Omega^p = \{\Omega_1^p, \dots, \Omega_s^p\}$  and  $\Omega^n = \{\Omega_1^n, \dots, \Omega_s^n\}$  are point information and normal information matrices, respectively. Information matrices have size of  $3 \times 3$  and are symmetric. The information matrices were introduced in [6] to enlarge the basin of convergence of the ICP algorithm. Let us briefly recall a general method for calculating point and normal information matrices.

The information matrices characterize the surface around a point  $q^i \in Q$  with its normal  $n_q^i$  and curvature  $\sigma_i$ . To compute the normal, the covariance matrix of the Gaussian distribution  $N_i(\mu_i, \Sigma_i)$  of all points lying in a sphere of radius  $r$  centered at the query point  $q^i$ , where the mean  $\mu_i$  and the covariance  $\Sigma_i$  are calculated as follows:

$$\mu_i = \frac{1}{|V_i|} \sum_{q^j \in V_i} q^j, \tag{38}$$

$$\Sigma_i = \frac{1}{|V_i|} \sum_{q^j \in V_i} (q^j - \mu_i)^t (q^j - \mu_i), \tag{39}$$

where  $V_i$  is the set of points composing the neighborhood of  $q^i$ . The eigenvalue decomposition of the matrix  $\Sigma_i$  is calculated as

$$\Sigma_i = \mathfrak{R} \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \mathfrak{R}^t, \tag{40}$$

where  $\lambda_1, \lambda_2, \lambda_3$  are the eigenvalues of  $\Sigma_i$  in ascending order,  $\mathfrak{R}$  is the matrix of eigenvectors. The normal  $n_i^q$  is taken as the smallest eigenvector and, if necessary, reoriented to the observer. The curvature  $\sigma_i$  is computed as

$$\sigma_i = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3}. \tag{41}$$

If the curvature  $\sigma_i$  is sufficiently small, then the matrix  $\Sigma_i$  is given as

$$\Sigma_i = \mathfrak{R} \begin{pmatrix} \varepsilon & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathfrak{R}^t, \tag{42}$$

where  $\varepsilon$  is a small coefficient. The point information matrix  $\Omega_i^p$  of point  $q_i$  is defined as

$$\Omega_i^p = (\Sigma_i)^{-1}. \tag{43}$$

If the curvature  $\sigma_i$  is small enough, the normal information matrix  $\Omega_i^n$  is set as follows:

$$\Omega_i^n = \mathfrak{R} \begin{pmatrix} \frac{1}{\varepsilon} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathfrak{R}^t, \tag{44}$$

otherwise, the matrix  $\Omega_i^n$  is equal to the identity matrix.

All information matrices are symmetric. Note that the point and normal information matrices can be individually selected as symmetric matrices for special point cloud types.

The functional  $J_\Omega$  (37) coincides with the functional (24) from [6] when the parameter  $\alpha$  is equal to zero. To solve the variational problem, the authors [6] use the iterative Gauss-Newton method based on quaternion parametrization of  $SO(3)$ , while we derive a closed form solution to the following variational problem: to find

$$arg \min_{R, T} J_\Omega(R, T). \tag{45}$$

The proposed solution is an approximation of the exact solution of the problem (45).

### 3.1 Functional $J$ in homogeneous coordinates

The functional  $J_\Omega(R, T)$  in homogeneous coordinates takes the form

$$J_\Omega(A) = \alpha \| R - I \|^2 + \sum_{i=1}^s (Ap^i - q^i)^t \Omega_i^{hp} (Ap^i - q^i) + \sum_{i=1}^s (An_i^p - n_i^q)^t \Omega_i^{hn} (An_i^p - n_i^q), \tag{46}$$

where

$$\begin{aligned}
 A &= \begin{pmatrix} r_{11} & r_{12} & r_{13} & T_1 \\ r_{21} & r_{22} & r_{23} & T_2 \\ r_{31} & r_{32} & r_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad p^i = (p_1^i \quad p_2^i \quad p_3^i \quad 1)^t, \quad q^i = (q_1^i \quad q_2^i \quad q_3^i \quad 1)^t, \\
 n_i^p &= ((n_i^p)_1 \quad (n_i^p)_2 \quad (n_i^p)_3 \quad 0)^t, \quad n_i^q = ((n_i^q)_1 \quad (n_i^q)_2 \quad (n_i^q)_3 \quad 0)^t, \\
 \Omega_i^{hp} &= \begin{pmatrix} (\Omega_i^p)_{11} & (\Omega_i^p)_{12} & (\Omega_i^p)_{13} & 0 \\ (\Omega_i^p)_{21} & (\Omega_i^p)_{22} & (\Omega_i^p)_{23} & 0 \\ (\Omega_i^p)_{31} & (\Omega_i^p)_{32} & (\Omega_i^p)_{33} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \Omega_i^{hn} = \begin{pmatrix} (\Omega_i^n)_{11} & (\Omega_i^n)_{12} & (\Omega_i^n)_{13} & 0 \\ (\Omega_i^n)_{21} & (\Omega_i^n)_{22} & (\Omega_i^n)_{23} & 0 \\ (\Omega_i^n)_{31} & (\Omega_i^n)_{32} & (\Omega_i^n)_{33} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \\
 T &= (T_1 \quad T_2 \quad T_3 \quad 0)^t, \quad i = 1, \dots, s.
 \end{aligned}$$

The matrices  $\Omega_i^{hp}$  and  $\Omega_i^{hn}$  are symmetric,  $i = 1, \dots, s$ . The matrix  $R$  is the  $3 \times 3$  submatrix of the matrix  $A$ . The variation problem (45) takes the following form: to find

$$\operatorname{arg\,min}_A J_\Omega(A). \tag{47}$$

### 3.2 Gradient of the functional $J$

Denote by  $J_1(A)$ ,  $J_2(A)$  and  $J_3(A)$  the following functionals:

$$\begin{aligned}
 J_1(A) &= \sum_{i=1}^s (Ap^i - q^i)^t \Omega_i^{hp} (Ap^i - q^i), \\
 J_2(A) &= \sum_{i=1}^s (An_i^p - n_i^q)^t \Omega_i^{hn} (An_i^p - n_i^q), \\
 J_3(A) &= \alpha \|R - I\|^2.
 \end{aligned} \tag{48}$$

Since  $J_\Omega(A) = J_1(A) + J_2(A) + J_3(A)$  then we have  $\nabla J_\Omega(A) = \nabla J_1(A) + \nabla J_2(A) + \nabla J_3(A)$ . At first, we consider the gradient  $\nabla J_1(A)$ . The functional  $J_1(A)$  can be rewritten as

$$\begin{aligned}
 J_1(A) &= \sum_{i=1}^s (Ap^i - q^i)^t \Omega_i^{hp} (Ap^i - q^i) \\
 &= \sum_{i=1}^s (p^i)^t A^t \Omega_i^{hp} Ap^i - 2 \sum_{i=1}^s (q^i)^t \Omega_i^{hp} Ap^i + \sum_{i=1}^s (q^i)^t \Omega_i^{hp} q^i.
 \end{aligned} \tag{49}$$

Let us look at the functional  $J_1(A + h)$ , where  $h$  is a small variation of the function  $A$ .

$$\begin{aligned}
 J_1(A + h) &= \sum_{i=1}^s (p^i)^t (A + h)^t \Omega_i^{hp} (A + h)p^i \\
 &\quad - 2 \sum_{i=1}^s (q^i)^t \Omega_i^{hp} (A + h)p^i + \sum_{i=1}^s (q^i)^t \Omega_i^{hp} q^i
 \end{aligned}$$

$$\begin{aligned}
 &= \sum_{i=1}^s (p^i)^t A^t \Omega_i^{hp} A p^i + \sum_{i=1}^s (p^i)^t h^t \Omega_i^{hp} A p^i + \sum_{i=1}^s (p^i)^t h^t \Omega_i^{hp} h p^i \\
 &\quad - 2 \sum_{i=1}^s (q^i)^t \Omega_i^{hp} A p^i - 2 \sum_{i=1}^s (q^i)^t \Omega_i^{hp} h p^i + \sum_{i=1}^s (q^i)^t \Omega_i^{hp} q^i. \tag{50}
 \end{aligned}$$

The difference between  $J_1(A + h)$  and  $J_1(A)$  is as follows:

$$J_1(A + h) - J(A) = 2 \sum_{i=1}^s (p^i)^t h^t \Omega_i^{hp} A p^i - 2 \sum_{i=1}^s (q^i)^t \Omega_i^{hp} h p^i + o(h). \tag{51}$$

**Proposition 1** *Let  $u$  and  $v$  be  $4 \times 1$  columns,  $M$  be a  $4 \times 4$  matrix. Then the following conditions holds:*

$$u^t M v = \text{tr}(M v u^t) = \text{tr}(v u^t M) = \langle v u^t, M^t \rangle. \tag{52}$$

Using Proposition 1, the linear part of (51) can be written as

$$\begin{aligned}
 &2 \sum_{i=1}^s (p^i)^t h^t \Omega_i^{hp} A p^i - 2 \sum_{i=1}^s (q^i)^t \Omega_i^{hp} h p^i \\
 &= 2 \sum_{i=1}^s \text{tr}(h^t \Omega_i^{hp} A p^i (p^i)^t) - 2 \sum_{i=1}^s \text{tr}(h p^i (q^i)^t \Omega_i^{hp}) \\
 &= 2 \text{tr} \left( \sum_{i=1}^s h^t \Omega_i^{hp} A p^i (p^i)^t \right) - 2 \text{tr} \left( \sum_{i=1}^s h p^i (q^i)^t \Omega_i^{hp} \right) \\
 &= 2 \text{tr} \left( \sum_{i=1}^s \Omega_i^{hp} A p^i (p^i)^t h^t \right) - 2 \text{tr} \left( \sum_{i=1}^s \Omega_i^{hp} q^i (p^i)^t h^t \right) \\
 &= \langle 2 \sum_{i=1}^s \Omega_i^{hp} A p^i (p^i)^t, h \rangle - \langle 2 \sum_{i=1}^s \Omega_i^{hp} q^i (p^i)^t, h \rangle \\
 &= \langle 2 \sum_{i=1}^s \Omega_i^{hp} A p^i (p^i)^t - 2 \sum_{i=1}^s \Omega_i^{hp} q^i (p^i)^t, h \rangle. \tag{53}
 \end{aligned}$$

We get that the gradient  $J_1(A)$  is expressed as

$$\nabla J_1(A) = 2 \sum_{i=1}^s \Omega_i^{hp} A p^i (p^i)^t - 2 \sum_{i=1}^s \Omega_i^{hp} q^i (p^i)^t. \tag{54}$$

In the same way, we can obtain the gradient of the functional  $J_2(A)$

$$\nabla J_2(A) = 2 \sum_{i=1}^s \Omega_i^{hn} A n_i^p (n_i^p)^t - 2 \sum_{i=1}^s \Omega_i^{hp} n_i^q (n_i^p)^t. \tag{55}$$

Finally, the gradient of the functional  $J_3(A)$  is given as

$$\nabla J_3(A) = 2\alpha \begin{pmatrix} r_{11} - 1 & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} - 1 & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} - 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{56}$$

### 3.3 Computation of the solution

Let us compute the extremal function of the functional  $J_\Omega$

$$\nabla J_\Omega(A) = \nabla J_1(A) + \nabla J_2(A) + \nabla J_3(A) = 0. \tag{57}$$

Let  $P^i$  and  $N^i$  be defined as follows:

$$P^i = p^i(p^i)^t, \quad N^i = n_i^p(n_i^p)^t, \tag{58}$$

where  $i = 1, \dots, s$ . We also define the data matrix  $M$  as

$$M = \sum_{i=1}^s \Omega_i^{hp} q^i (p^i)^t - \sum_{i=1}^s \Omega_i^{hn} n_i^q (n_i^p)^t. \tag{59}$$

Then the formula (57) can be rewritten as

$$\sum_{i=1}^s \Omega_i^{hp} A P^i + \sum_{i=1}^s \Omega_i^{hn} A N^i + \nabla J_3(A) = M. \tag{60}$$

Let us rewrite the system of linear equations (60) in the matrix form as

$$B A' = M', \tag{61}$$

where

$$\begin{aligned} M' &= (M_{11} + \alpha \quad M_{12} \quad M_{13} \quad M_{14} \quad M_{21} \quad M_{22} + \alpha \\ &\quad M_{23} \quad M_{24} \quad M_{31} \quad M_{32} \quad M_{33} + \alpha \quad M_{34})^t, \\ A' &= (A_{11} \quad A_{12} \quad A_{13} \quad A_{14} \quad A_{21} \quad A_{22} \quad A_{23} \quad A_{24} \quad A_{31} \quad A_{32} \quad A_{33} \quad A_{34})^t, \end{aligned}$$

$B$  is a  $12 \times 12$  matrix. In order to determine the entries of  $B$ , we first define

$$H_{klmn} = \sum_{i=1}^s (\Omega_i^{hp})_{kl} (P^i)_{mn}, \quad G_{klmn} = \sum_{i=1}^s (\Omega_i^{hn})_{kl} (N^i)_{mn}, \tag{62}$$

where  $k = 1, \dots, 3, n = 1, \dots, 4, l = 1, \dots, 3$  and  $m = 1, \dots, 4$ . Auxiliary  $12 \times 12$  matrix  $\hat{B}$  is defined as

$$\hat{B}_{4(k-1)+n, 4(l-1)+m} = H_{klmn} + G_{klmn}. \tag{63}$$

The matrix  $B$  is calculated as

$$B = \hat{B} + \text{diag}(\alpha \quad \alpha \quad \alpha \quad 0 \quad \alpha \quad \alpha \quad \alpha \quad 0 \quad \alpha \quad \alpha \quad \alpha \quad 0) \tag{64}$$

**Proposition 2** *The matrix  $B$  is symmetric.*

Proposition 2 follows from

$$H_{klmn} = \sum_{i=1}^s (\Omega_i^{hp})_{kl} (P^i)_{mn} = \sum_{i=1}^s (\Omega_i^{hp})_{lk} (P^i)_{nm} = H_{lknm}. \tag{65}$$

**Corollary 1** *In general, to calculate the matrix  $B$ , it is necessary to know 144 elements of  $H_{klmn}$ . Proposition 2 reduces this number to 78 (the same for  $G_{klmn}$ ). Proposition 2 also helps to numerically solve the linear system (62) using methods with self-conjugate matrices.*

### 3.4 Projection on $SO(3)$ and recalculation of translation vector

Let  $\hat{A}$  be a  $3 \times 3$  left-upper (affine) submatrix of the matrix  $A$ . Let us consider singular value decomposition  $UDV^t$  of the matrix  $\hat{A}$ . The projection  $R_*$  of  $\hat{A}$  to the manifold  $SO(3)$  is the matrix  $USV^t$ , where  $S$  is given by the formula (30).

The translation vector  $T_*$  is computed as

$$T_* = W \sum_{i=1}^s \Omega_i^{hp} (q^i - R_* p^i), \quad (66)$$

where

$$W = \left( \sum_{i=1}^s \Omega_i^{hp} \right)^{-1}.$$

The orthogonal transformation  $(R_*, T_*)$  is the proposed approximation of the exact solution to the variational problem (45).

### 3.5 Summary of the algorithm

The input data for the algorithm are the point clouds  $P$  and  $Q$ , the correspondence between clouds, the point and normal information matrices  $\Omega^p$  and  $\Omega^n$ . The algorithm consists of the following steps:

- (1) represent the functional  $J_\Omega(R, t)$  (37) as the functional  $J_\Omega(A)$  (46) in homogeneous coordinates;
- (2) compute the gradient  $\nabla J_\Omega(A)$  of the functional  $J_\Omega(A)$ ;
- (3) compute extremal matrix  $A$  from the equation  $\nabla J_\Omega(A) = 0$  (57);
- (4) extract a left-upper  $3 \times 3$  submatrix  $\hat{A}$  from the matrix  $A$ ;
- (5) compute orthogonal matrix  $R_*$  as the projection  $\hat{A}$  to submanifold  $SO(3)$  (see detailed description in Section 3.4);
- (6) compute translation vector  $T_*$  by the formula (66).

## 4 Computer simulation

In this section, with the help of computer simulation, we show the performance of the proposed and NICP [6] algorithms in terms of the following criteria: convergence rate (the frequency of convergence of ICP algorithm to a correct solution for given conditions); total processing time; processing time for solving a variation problem; number of iterations (for ICP algorithm). The tested algorithms use the same parameters of the ICP algorithm, with the exception of the parameters of variational problems. Implementation of the NICP variational problem is taken from [16]. The same level of parallel optimization of the NICP and proposed algorithms for software implementations are used. All values of parameters were manually selected for the best performance of the all tested algorithms.

The experiments are organized as follows. Orthogonal geometric transformation given by a known matrix is applied to a source point cloud. Source and transformed clouds are input to a tested ICP algorithm. The ICP algorithm converges if the reconstructed transformation matrix coincides with the original matrix with a given accuracy.

The program realization of the NICP variational problem allows to vary the number of iterations of the Gauss-Newton method. We consider here the variants with one iteration (of the Gauss-Newton method)  $\lambda\_NICP\_1$  and  $\Omega\_NICP\_1$ , with 15 iterations (of the Gauss-Newton method)  $\lambda\_NICP\_15$ .

We denote by  $\lambda\_MH-ICP$  the ICP algorithm variant based on the variation problem (5), by  $\lambda\_MH-RICP$  the ICP variant based on the variational problem (7). The NICP functional (24) in [6] coincides with functional (4) when all point information matrices are identity and all normal information matrices are  $\lambda I$ , where  $I$  is identity matrix

$$\Omega_i^p = I, \quad \Omega_i^n = \lambda I, \tag{67}$$

where  $i = 1, \dots, s$ . We denote by  $\lambda\_NICP\_1$  and  $\lambda\_NICP\_15$  the respective variants of the NICP algorithm with restriction (67), by  $\Omega\_NICP\_1$  the respective variant of the NICP algorithm with nontrivial information matrices.

We denote by  $\Omega\_MH-RICP$  the ICP algorithm variant based on the variation problem (45), by  $\Omega\_MH-ICP$  the ICP variant based on the same variational problem when the parameter  $\alpha$  is equal to zero. When comparing the performance of  $\Omega\_MH-RICP$  with that of  $\Omega\_NICP\_1$ , the best information matrices are selected for each algorithm and each 3D model.

### 4.1 Separate examples

#### 4.1.1 Stanford bunny

3D model of the Stanford bunny [17] consists of 34817 points. The resultant point cloud Q is obtained from the cloud P by an orthogonal transformation, defined by the following matrix  $A_{true}$ :

$$A_{true} = \begin{pmatrix} 0.554098 & 0.445902 & 0.702956 & 0.986970 \\ 0.445902 & 0.554098 & -0.702956 & 2.070842 \\ -0.702956 & 0.702956 & 0.108196 & 2.958520 \\ 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{pmatrix}.$$

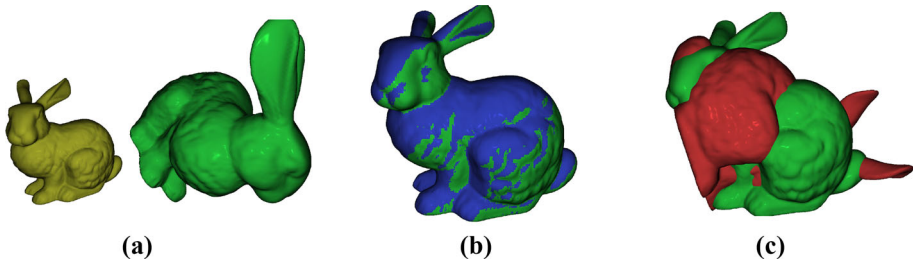
For both algorithms  $\lambda = 0.3$ . The recovered matrix by the proposed  $\lambda\_MH-RICP$  algorithm is

$$A_{\lambda\_MH-RICP} = \begin{pmatrix} 0.554098 & 0.445902 & 0.702956 & 0.986970 \\ 0.445902 & 0.554098 & -0.702956 & 2.070842 \\ -0.702956 & 0.702956 & 0.108196 & 2.958520 \\ 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{pmatrix}.$$

The recovered matrix by the  $\lambda\_NICP\_1$  algorithm is

$$A_{\lambda\_NICP\_1} = \begin{pmatrix} 0.326606 & 0.369834 & -0.869799 & 1.138148 \\ 0.384461 & 0.788708 & 0.479718 & 1.539322 \\ 0.863434 & -0.491083 & 0.115410 & 4.719121 \\ 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{pmatrix}.$$

Figure 1a shows a source cloud P (yellow color) and the resultant cloud Q (green color) obtained from P by the geometrical transformation  $A_{true}$ . Figure 1b shows the alignment result of the clouds with the  $\lambda\_MH-RICP$  algorithm. Figure 1c shows alignment result of the clouds with the  $\lambda\_NICP\_1$  algorithm.



**Fig. 1** **a** Test cloud P (yellow), resultant cloud Q (green); **b** alignment result of the clouds with  $\lambda_{MH-RICP}$  algorithm; **c** alignment result with  $\lambda_{NICP_1}$  algorithm. (Color figure online)

#### 4.1.2 Armadillo

3D model of Armadillo [17] consists of 21259 points. The resultant point cloud Q is obtained from the cloud P by the orthogonal transformation, defined by an following matrix  $A_{true}$ :

$$A_{true} = \begin{pmatrix} 0.589745 & -0.410255 & 0.695623 & 1.017242 \\ -0.410255 & 0.589745 & 0.695623 & 1.997038 \\ -0.695623 & -0.695623 & 0.179491 & 3.022741 \\ 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{pmatrix}.$$

For both algorithms  $\lambda = 0.3$ . The recovered matrix by the proposed  $\lambda_{MH-RICP}$  algorithm is

$$A_{\lambda_{MH-RICP}} = \begin{pmatrix} 0.589745 & -0.410255 & 0.695623 & 1.017242 \\ -0.410255 & 0.589745 & 0.695623 & 1.997038 \\ -0.695623 & -0.695623 & 0.179490 & 3.022741 \\ 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{pmatrix}.$$

The recovered matrix by the  $\lambda_{NICP_1}$  algorithm is

$$A_{\lambda_{NICP_1}} = \begin{pmatrix} 0.805599 & 0.358720 & 0.471518 & 0.325207 \\ 0.285274 & 0.462664 & -0.839381 & 2.300575 \\ -0.519257 & 0.810717 & 0.270388 & 1.718160 \\ 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{pmatrix}.$$

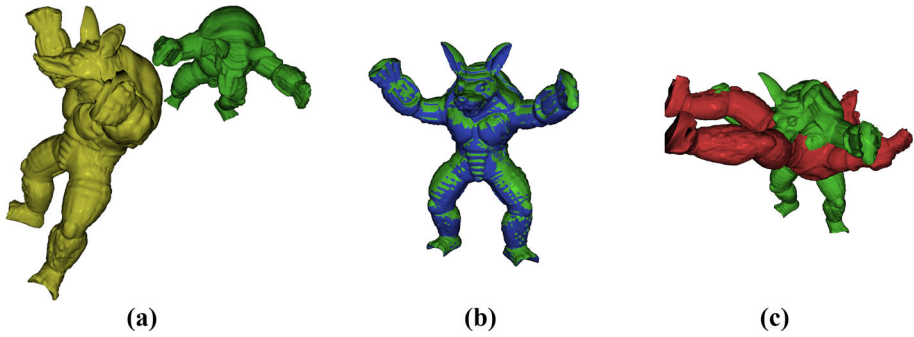
Figure 2a shows a source cloud P (yellow color) and the resultant cloud Q (green color) obtained from P by the geometrical transformation  $A_{true}$ . Figure 2b shows the alignment result of the clouds with the  $\lambda_{MH-RICP}$  algorithm. Figure 2c shows the alignment result of the clouds with the  $\lambda_{NICP_1}$  algorithm.

#### 4.2 Influence of regularization to the performance of the proposed algorithms

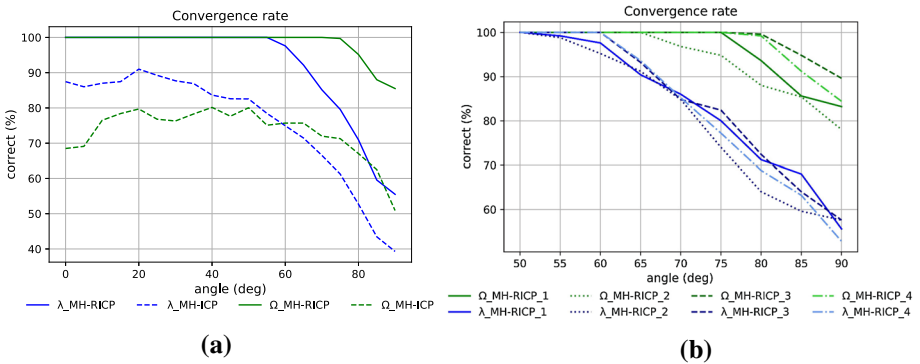
The purpose of this section is to study the influence of the proposed regularization on the performance of different ICP algorithms with poor correspondence between the source and target point clouds at the first iterations.

We consider here  $\lambda_{MH-ICP}$ ,  $\lambda_{MH-RICP}$ ,  $\Omega_{MH-RICP}$  and  $\Omega_{MH-ICP}$  algorithms. For the first five iterations of ICP algorithm with regularization are used the following values of  $\alpha$ : 10000; 5000; 2500; 1250; 625. And for the following iterations, zero values of the parameter  $\alpha$  are used.





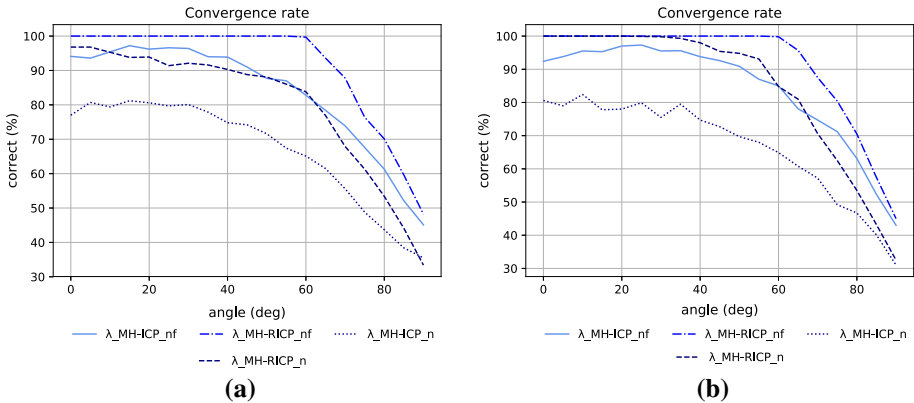
**Fig. 2** **a** Test cloud P (yellow), resultant cloud Q (green); **b** alignment result of the clouds with  $\lambda\_MH-RICP$  algorithm; **c** alignment result with  $\lambda\_NICP\_1$  algorithm. (Color figure online)



**Fig. 3** **a** The convergence rate of  $\lambda\_MH-RICP$  (blue),  $\lambda\_MH-ICP$  (blue dashed line),  $\Omega\_MH-RICP$  (green),  $\Omega\_MH-ICP$  (green dashed line) algorithms; **b** the convergence rate of  $\lambda\_MH-RICP\_1$ ,  $\lambda\_MH-RICP\_2$ ,  $\lambda\_MH-RICP\_3$ ,  $\lambda\_MH-RICP\_4$  algorithms (blue lines) and  $\Omega\_MH-RICP\_1$ ,  $\Omega\_MH-RICP\_2$ ,  $\Omega\_MH-RICP\_3$ ,  $\Omega\_MH-RICP\_4$  algorithms (green lines). (Color figure online)

In this experiment, a 3D model of the Stanford bunny consisting of 1250 points is used. Note that the point cloud  $P$  lies in a centered cube with the edge length of 2. Let us set the value of the rotation angle. We fix the value of the rotation angle. We take a random, uniformly distributed direction vector that defines a line containing the origin of the coordinate system. This line is the axis of rotation at a fixed angle. Also, the components of the translation vector are a random variable uniformly distributed in the interval [8, 10]. The synthesized geometric transformation matrix (original matrix) is applied to the source cloud of points  $P$ . The resulting cloud is  $Q$ . The tested algorithms are applied to the clouds  $P$  and  $Q$ . We say that the ICP algorithm converges to true data if the reconstructed transformation matrix coincides with the original matrix with accuracy up to three digits after the decimal point. To guarantee statistically correct results, 1000 trials for each fixed rotation angle are carried out. The rotation angle varies from 0 to 90 degrees with a step of 5 degrees. Figure 3a shows the convergence rate of  $\lambda\_MH-RICP$  (blue),  $\lambda\_MH-ICP$  (blue dashed line),  $\Omega\_MH-RICP$  (green),  $\Omega\_MH-ICP$  (green dashed line) algorithms.

**Remark 5** The convergence rate of the algorithms with regularization ( $\lambda\_MH-RICP$  and  $\Omega\_MH-RICP$ ) is much better than that of the corresponding algorithms without regular-



**Fig. 4** The convergence rate of  $\lambda\_MH\text{-ICP\_nf}$ ,  $\lambda\_MH\text{-RICP\_nf}$ ,  $\lambda\_MH\text{-ICP\_n}$  and  $\lambda\_MH\text{-RICP\_n}$  algorithms in the case of: **a** Gaussian noise; **b** impulse noise

ization ( $\lambda\_MH\text{-ICP}$  and  $\Omega\_MH\text{-ICP}$ ). Therefore, in further experiments, only the proposed algorithms with regularization will be used.

Now we study the convergence of the ICP algorithms with regularization for a different choice of  $\alpha$  at the first iterations. Four following variants of choosing the values of  $\alpha$  are tested:

- (1) 10000, 5000, 2500, 1250, 625, 0, ..., ( $\lambda\_MH\text{-RICP\_1}$  and  $\Omega\_MH\text{-RICP\_1}$ );
- (2) 10000 for the first ten iterations, 0, ..., ( $\lambda\_MH\text{-RICP\_2}$  and  $\Omega\_MH\text{-RICP\_2}$ );
- (3) 1000000,  $\frac{1000000}{2}$ ,  $\frac{1000000}{4}$ ,  $\frac{1000000}{8}$ , ..., ( $\lambda\_MH\text{-RICP\_3}$  and  $\Omega\_MH\text{-RICP\_3}$ );
- (4) 1000000 for the first ten iterations, 0, ..., ( $\lambda\_MH\text{-RICP\_4}$  and  $\Omega\_MH\text{-RICP\_4}$ ).

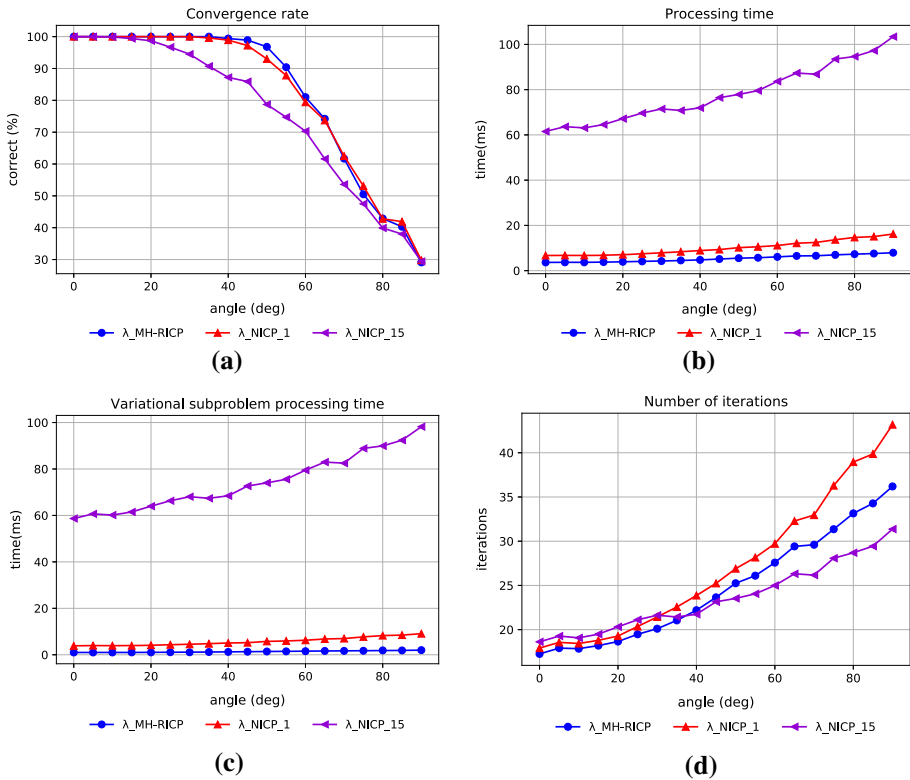
Figure 3b shows the convergence rate of the tested algorithms depending on the rotation angle between point clouds. Note that the third variant of choosing the values of  $\alpha$  yields the best convergence rate. However, we use further the first variant because it gives suitable convergence and requires the minimal processing time among the tested algorithms.

Next, we examine the performance of the proposed algorithm with regularization in the presence of outliers modeled by additive and impulse noise. As the source point cloud, a 3D model of the Stanford bunny is used (in the original coordinate system, the points lie in a centered cube with the edge length of 2). Clouds  $P$  and  $Q$  are independently corrupted noise. The  $L_2$  norm of the residual of the matrices of the true geometric transformation and the result of the ICP algorithm is calculated. If the norm value is less than the threshold of 0.2, then we say that the ICP algorithm finds the correct transformation.

First, outliers are modeled by additive Gaussian noise. Each coordinate of each point is distorted by adding a Gaussian random variable with a zero-mean and a standard deviation of  $\sigma = 0.12$ .

Second, outliers are modeled by impulse noise. The probability of distortion of a point by impulse noise is 0.1. Each coordinate of the distorted point is changed by adding a random variable uniformly distributed in the interval  $[-0.4; 0.4]$ .

Denote by  $\lambda\_MH\text{-RICP\_nf}$  and  $\lambda\_MH\text{-ICP\_nf}$  the respective algorithms applied to noise-free clouds. Denote by  $\lambda\_MH\text{-RICP\_n}$  and  $\lambda\_MH\text{-ICP\_n}$  the respective algorithms applied to noisy clouds. Figure 4a, b show the convergence rate of the tested algorithms in the case of additive Gaussian noise and impulse noise, respectively.



**Fig. 5** The performance of the tested algorithms: **a** convergence rate of  $\lambda_{NICP\_1}$  (red),  $\lambda_{NICP\_15}$  (purple) and  $\lambda_{MH-RICP}$  (blue); **b** total processing time of  $\lambda_{NICP\_1}$  (red),  $\lambda_{NICP\_15}$  (purple) and  $\lambda_{MH-RICP}$  (blue); **c** processing time for solving the variational problem of  $\lambda_{NICP\_1}$  (red),  $\lambda_{NICP\_15}$  (purple) and  $\lambda_{MH-RICP}$  (blue); **d** number of ICP iterations of  $\lambda_{NICP\_1}$  (red),  $\lambda_{NICP\_15}$  (purple) and  $\lambda_{MH-RICP}$  (blue). (Color figure online)

One can observe that the algorithm  $\lambda_{MH-RICP}$  is much robust to outliers than the same algorithm without regularization  $\lambda_{MH-ICP}$ .

### 4.3 Comparison of $\lambda_{NICP\_1}$ and $\lambda_{NICP\_15}$ algorithms

Since the performance of the NICP algorithm depends on the number of iterations of the Gauss-Newton method, in our experiments we compare the performance of the algorithm with 1 ( $\lambda_{NICP\_1}$ ) and 15 ( $\lambda_{NICP\_15}$ ) iterations with respect to convergence rate, total processing time, processing time for solving a variation problem, and number of iterations (for ICP algorithm). Additionally, the proposed algorithm  $\lambda_{MH-RICP}$  is also tested. The value of  $\lambda$  for the tested algorithms is equal to 0.3.

Figure 5 shows the performance of the algorithms with respect to the used criteria.

One can observe that the performance of the  $\lambda_{NICP\_1}$  algorithm is better in terms of convergence rate, total running time, processing time for solving a variation problem than that of the  $\lambda_{NICP\_15}$  algorithm. So, in further experiments, only the NICP algorithm variants utilized one iteration of the Gauss-Newton method will be used.

## 4.4 Comparison of the proposed algorithms and NICP

The Sect. 4.4 is the main part of the Sect. 4. Now we compare the performance of the proposed algorithms and the  $\lambda\_NICP\_1$ ,  $\Omega\_NICP\_1$  algorithms.

### 4.4.1 Stanford bunny

The conditions of the experiment are described in the Sect. 4.2.

In this experiment,  $\lambda\_MH-RICP$  and  $\Omega\_MH-RICP$  use variable values of the regularization parameter  $\alpha$ . For the first five iterations of the ICP algorithm the parameter  $\alpha$  takes the following values of: 10000; 5000; 2500; 1250; 625. For the next iterations, zero values of the parameter are used.

The point and normal information matrices are computed by formulas (43) and (44), respectively.

If the curvature (formula (41)) is more or equal 0.02 we use diagonal matrices (instead diagonal matrix in formula (40))  $diag(1.5, 1, 1, 0)$  and  $diag(0.005, 1, 1, 0)$  (as normal information matrices, formula (44)) for  $\Omega\_MH-RICP$  algorithm,  $diag(1, 1, 1, 0.0)$  (instead diagonal matrix in formula (40)) and  $diag(0.0005, 0.1, 0.1, 0)$  (as normal information matrices, formula (44)) for  $\Omega\_NICP\_1$ .

If the curvature is less 0.02 we use diagonal matrices  $diag(1, 1, 1, 0)$  (instead diagonal matrix in formula (42)) and  $diag(0.1, 1, 1, 0)$  (instead diagonal matrix in formula (44)) for  $\Omega\_MH-RICP$  algorithm,  $diag(10, 1, 1, 0)$  (instead diagonal matrix in formula (42)) and  $diag(0.1, 0.01, 0.01, 0)$  (instead diagonal matrix in formula (44)) for  $\Omega\_NICP\_1$ . The parameters of point and normal information matrices were manually selected for best performance for the all considered algorithms.

Figure 6 shows the performance of  $\lambda\_NICP\_1$  (red),  $\Omega\_NICP\_1$  (yellow),  $\lambda\_MH-RICP$  (blue) and  $\Omega\_MH-RICP$  (green) algorithms for: (a) convergence rate; (b) total processing time; (c) processing time for solving the variational problem; (d) ICP iterations numbers.

One can observe that the  $\lambda\_MH-RICP$  algorithm outperforms the  $\lambda\_NICP\_1$  algorithm with respect all the criteria used, and of the performance of the  $\Omega\_MH-RICP$  algorithm is better than that of the  $\Omega\_NICP\_1$  algorithm in terms of convergence rate, total processing time, and processing time for solving a variation problem.

### 4.4.2 Armadillo

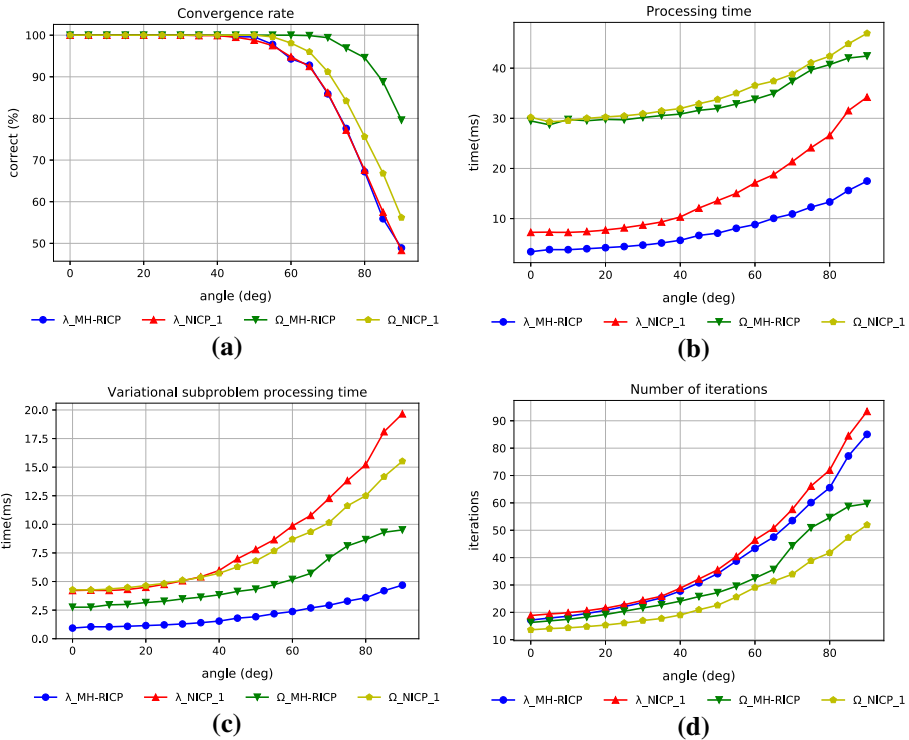
The conditions of the experiment are described in the Sect. 4.2.

In this experiment,  $\lambda\_MH-RICP$  and  $\Omega\_MH-RICP$  use variable values of the regularization parameter  $\alpha$ . For the first five iterations of the ICP algorithm the parameter  $\alpha$  takes the following values of: 10000; 5000; 2500; 1250; 625. For the next iterations, zero values of the parameter are used.

The point and normal information matrices are computed by formulas (43) and (44), respectively.

If the curvature (formula (41)) is more or equal 0.02 we use diagonal matrices (instead diagonal matrix in formula (40))  $diag(0.5, 1, 1, 0)$  and  $diag(0.005, 1, 1, 0)$  (as normal information matrices, formula (44)) for  $\Omega\_MH-RICP$  algorithm,  $diag(0.15, 0.1, 0.1, 0)$  (instead diagonal matrix in formula (40)) and  $diag(0.1, 1, 1, 0)$  (as normal information matrices, formula (44)) for  $\Omega\_NICP\_1$ .

If the curvature is less 0.02 we use diagonal matrices  $diag(1, 1, 1, 0)$  (instead diagonal matrix in formula (42)) and  $diag(0.1, 1, 1, 0)$  (instead diagonal matrix in formula (44))



**Fig. 6** The performance of the tested algorithms: **a** convergence rate of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green); **b** total processing time of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green); **c** processing time for solving the variational problem of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green); **d** number of ICP iterations of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green). (Color figure online)

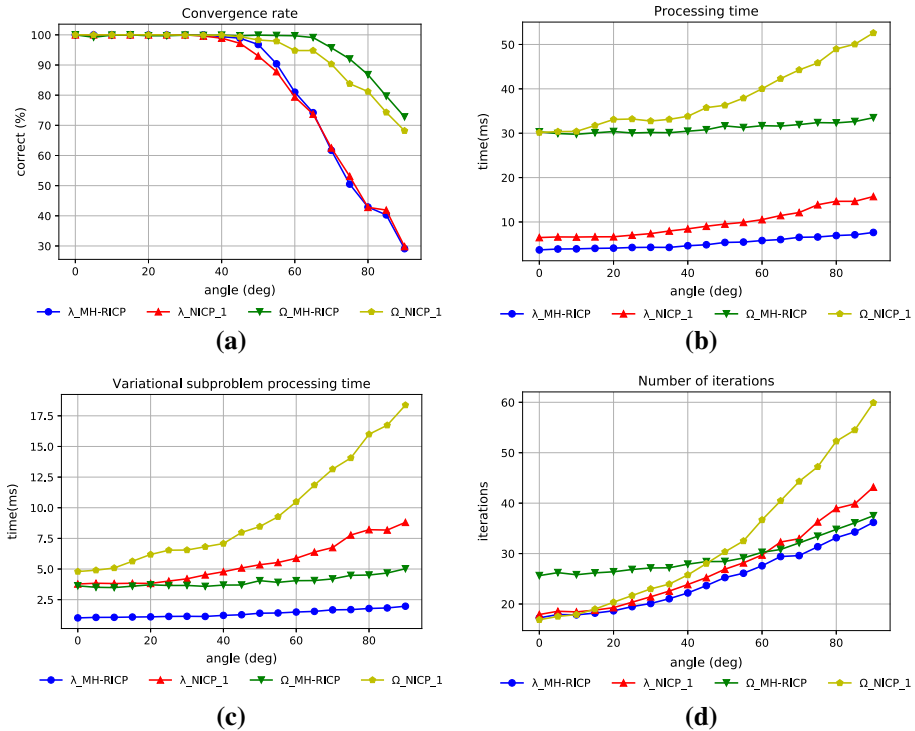
for  $\Omega_{MH-RICP}$  algorithm,  $diag(10, 1, 1, 0)$  (instead diagonal matrix in formula (42)) and  $diag(0.1, 0.01, 0.01, 0)$  (instead diagonal matrix in formula (44)) for  $\Omega_{NICP\_1}$ . The parameters of point and normal information matrices were manually selected for best performance for the all considered algorithms.

Figure 7 shows the performance of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green) algorithms for:(a) convergence rate; (b) total processing time; (c) processing time for solving the variational problem; (d) ICP iterations numbers.

One can observe that the  $\lambda_{MH-RICP}$  algorithm outperforms the  $\lambda_{NICP\_1}$  algorithm with respect all the criteria used, and of the performance of the  $\Omega_{MH-RICP}$  algorithm is better than that of the  $\Omega_{NICP\_1}$  algorithm in terms of convergence rate, total processing time, and processing time for solving a variation problem.

**Remark 6** The experiments show that the proposed  $\lambda_{MH-RICP}$  algorithm has a slightly better convergence rate compared to the  $\lambda_{NICP\_1}$  algorithm and is significantly faster, and the proposed  $\Omega_{MH-RICP}$  algorithm has a better convergence rate than the  $\Omega_{NICP\_1}$  algorithm and also faster than it.

**Remark 7** The proposed algorithms have some disadvantages. So the proposed regularization increases the processing time of the ICP algorithm. In addition, the  $\Omega_{MH-RICP}$  and



**Fig. 7** The performance of the tested algorithms: **a** convergence rate of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green); **b** total processing time of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green) algorithms; **c** processing time for solving the variational problem of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green); **d** number of ICP iterations of  $\lambda_{NICP\_1}$  (red),  $\Omega_{NICP\_1}$  (yellow),  $\lambda_{MH-RICP}$  (blue) and  $\Omega_{MH-RICP}$  (green). (Color figure online)

$\Omega_{MH-RICP}$  algorithms, as well as the  $\Omega_{NICP\_1}$  algorithm, are sensitive to the choice of point and normal information matrices.

### 5 Conclusion

In this paper, we proposed a regularization approach to 3D point cloud registration. New variants of the variational problem of the ICP algorithm were formulated. A closed-form exact solution for orthogonal registration of point clouds based on the generalized point-to-point ICP algorithm and a closed-form approximate solution to the NICP variational problem were derived. We also proposed robust ICP algorithms by adding a regularization term to common variational functionals that are able to find a suitable transformation between point clouds with poor correspondence. With the help of extensive computer simulation, we showed that the proposed algorithms yield good convergence to a suitable transformation, even when the correspondence of points is estimated incorrectly at the first iterations. The proposed approach can be applied to various ICP variational functionals.

## References

1. Makovetskii, A., Voronin, S., Kober, V., Voronin, A.: A generalized point-to-point approach for orthogonal transformations. *CCIS* **1090**, 217–231 (2019)
2. Besl, P., McKay, N.: A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(2), 239–256 (1992)
3. Chen, Y., Medioni, G.: Object modelling by registration of multiple range images. *Image Vis. Comput.* **10**(3), 145–155 (1992)
4. Turk, G., Levoy, M., Zippered polygon meshes from range images. In: *Computer Graphics Proceeding. Annual Conference Series. ACM SIGGRAPH*, pp. 311–318 (1994)
5. Segal, A., Haehnel, D., Thrun, S.: Generalized-ICP. In: *Robotics: Science and Systems* **5**, 161–168 (2010)
6. Serafin, J., Grisetti, G.: Using extended measurements and scene merging for efficient and robust point cloud registration. *Robot. Auton. Syst.* **92**, 91–106 (2017)
7. Du, S., Zheng, N., Meng, G., Yuan, Z.: Affine registration of point sets using ICP and ICA. *IEEE Signal Process. Lett.* **15**, 689–692 (2008)
8. Van der Sande, C., Soudarissanane, S., Khoshelham, K.: Assessment of relative accuracy of AHN-2 laser scanning data using planar features. *Sensors* **10**(9), 8198–8214 (2010)
9. Makovetskii, A., Voronin, S., Kober, V., Tihonkih, D.: Affine registration of point clouds based on point-to-plane approach. *Procedia Eng.* **201**, 322–330 (2017)
10. Horn, B.: Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. A* **4**(4), 629–642 (1987)
11. Horn, B., Hilden, H., Negahdaripour, S.: Closed-form solution of absolute orientation using orthonormal matrices. *J. Opt. Soc. Am. Ser. A* **5**(7), 1127–1135 (1988)
12. Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. *IEEE-TPAMI* **13**(4), 376–380 (1991)
13. Magnusson, M., Lilienthal, A., Duckett, T.: Scan registration for autonomous mining vehicles using 3D-NDT. *J. Field Robot* **24**(10), 803–827 (2007)
14. Newcombe, R., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A., Kohli, P., Shotton, J., Hodges, S., Fitzgibbon, A.: KinectFusion: Realtime dense surface mapping and tracking. In: *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)* (2011)
15. Makovetski, A., Voronin, S., Kober, V., Voronin, A.: A non-iterative method for approximation of the exact solution to the point-to-plane variational problem for orthogonal transformation. *Math. Methods Appl. Sci.* **41**(18), 9218–9230 (2018)
16. Normal Iterative Closest Point (NICP), Algorithm C++ Library. <http://github.com/yorsh87/nicp>. Accessed Mar 2020
17. Stanford University Computer Graphics Laboratory. The Stanford 3D Scanning Repository website. <http://graphics.stanford.edu/data/3Dscanrep>. Accessed Mar 2020

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.