Iterative Closest Point Problem: A Tensorial Approach to Finding the Initial Guess

Adrian Burlacu, Alexandru Cohal, Simona Caraiman and Daniel Condurache
Faculty of Automatic Control and Computer Engineering
Technical University "Gheorghe Asachi" of Iasi, Romania
Emails: { aburlacu}@ac.tuiasi.ro

***Abstract*—The iterative closest point (ICP) problem is very important in multiple fields such as robotics, machine vision, automotive or assistive technologies. The problem is to find the optimal transformation that can align two sets of 3D points. Even if in the recent year new variations of ICP were proposed, the algorithm may get trapped in local minima due to the non-convexity of the problem. This issue may be overcome if the initial guess is chosen as close as possible to the true solution. In this paper a tensorial based method is proposed for choosing the initial guess of the iterative closest point problem. This new approach is strongly connected with the parameters that can be used to describe the displacement of rigid bodies. Using an isomorphism between the special Euclidean group SE_3 and the orthogonal dual tensors group $\underline{S\mathbb{O}}_3$, a detailed procedure is described on how to compute the initial guess for the iterative closest point problem. An evaluation of the proposed method is done using a Matlab framework that implements the ICP algorithm.**

Keywords: *Iterative closest point, dual vectors, orthogonal dual tensors, initial guess*

I. Introduction

The Iterative Closest Point algorithm has been widely studied and developed, becoming one of the most used methods for aligning multi-dimensional models based on an initial estimation. The range of applications that use ICP vary from automotive to visually impaired assistive devices.

The first formulation of this algorithm was presented in [1], where the problem of constructing a complete model of a physical object was studied. Having two views $\mathbf{P}$ and $\mathbf{Q}$ of the same surface, the scope was to find a rigid transformation $\mathbf{H}$ such that the given views to be brought into coincidence (equation (1)). The presented solution of this problem is based on an iterative algorithm which minimizes a least square error measure. Point-to-point matches are not required due to the fact that the distance from points to planes is minimized. There are two main steps followed by this registration algorithm:

Step 1. Select a set of N control points $p_i \in \mathbf{P}$ (which can be picked on a regular grid because it is not needed to represent meaningful surface features). Then compute the surface normals n_{p_i} at these points.

Step 2. For each iteration k, repeat the following two steps until the procedure converges (an initial approximate transformation $\mathbf{H}^0$ is considered to be known)

Step 2.1. Apply the transformation $\mathbf{H}^{k-1}$ for each control point p_i and its surface normal n_{p_i} to obtain p_i' and n_{p_i}'. Find the intersection point q_i^k between the surface $\mathbf{Q}$ and the normal line defined by p_i' and n_{p_i}'. Determine the tangent plane S_i^k of $\mathbf{Q}$ at q_i^k.

Step 2.2. Find the transformation $\mathbf{H}$ which minimizes e^k expressed in equation (2) (where d_S is the signed distance from a point to a plane) with a least square method. The solution at iteration k is shown in (3):

$$\forall p \in \mathbf{P}, \exists q \in \mathbf{Q} \text{ such that } \|\mathbf{H} \cdot p - q\| = 0, \tag{1}$$

$$e^k = \sum_{i=1}^{N} d_S(\mathbf{H} \circ \mathbf{H}^{k-1} \cdot p_i, S_i^k)^2, \tag{2}$$

$$\mathbf{H}^k = \mathbf{H} \circ \mathbf{H}^{k-1}. \tag{3}$$

A version of the ICP algorithm based on quaternions is presented in [2]. This method is preffered only for the 2D and 3D cases, whereas for multi-dimensions, the Singular Value Decomposition (SVD) algorithm has to be substituted. In this paper an accelerated ICP algorithm is also described. A minor variation on the basic line search methods of multivariate unconstrained minimization leads to a quicker convergence to a local minimum.

In contrast with this algorithm, the one presented in [3] is capable of dealing with gross outliers in the given data, appearance and disappearance of curves in consecutives sets and occlusion. In this approach, *k-D tree* data structure is used for optimization in order to speed up the searching process. Constraints (e.g. distance continuity, orientation consistency) are imposed such that pairing of spurious points is avoided.

Many ICP variants are enumerated, classified and evaluated in [4] in order to find out their strong points and to combine them in a high-speed version suitable for real-time usage. An optimization of the classic ICP algorithm regarding the robustness issue is described in [5] where only a certain number of square errors are minimized after a sorting procedure. This method is called Least Trimmed Squares and is consistently used in all the phases of the operation. A hybrid approach between the original version of the ICP algorithm and a genetic algorithm is presented in [6]. The purpose of the genetic procedure is to perform a preliminary rough search in order to avoid local minimum points and to reduce the solution space.

In [7] it is introduced a generalization based on Lie groups for affine registration of multi-dimensional point sets. This

978-1-5090-2720-0/16/$31.00 ©2016 IEEE

problem is solved by reduction to a quadratic programming problem which assures the convergence to a local minimum for any initial set of parameters provided. The scale factor is taken into account in [8] and another version of the ICP algorithm is described. The solution is based on the integration of a scale matrix with boundaries into the Least Squares procedure of the classic algorithm. Due to the fact that the original ICP algorithm may fail to register point sets with noise, an approach based on expectation maximization principle to solve this problem by using a Gaussian probability model is introduced in [9]. The ideas from [8] and [9] are merged in [10] in order to develop a version of ICP algorithm which takes into account both the scale factor and the noise in the point sets.

In order to improve the robustness of registration and reduce the variety of rotation (which may lead to algorithms failure), the ideas presented in [11] and [12] are based on a Least Squares registration model with inequality constraints which has the scope of bounding the rotation angle for the case of 3D points sets and 2D point sets, respectively.

In this paper we present a new approach for computing the initial guess for the ICP problem. The proposed method is strongly connected with the parameters that can be used to describe the displacement of rigid bodies. Different approaches can be considered for rigid-body displacement parameterization. Recently, orthogonal dual tensors [13], [14] proved to be a complete free of coordinates tool that can be used to compute rigid body displacement and motion parameters. The present research makes use of the isomorphism between the special euclidean group SE_3 and the orthogonal dual tensors group $\underline{S\mathbb{O}}_3$, which allows the development of a new technique for finding the initial guess of the ICP problem. We give a detailed procedure that evaluates the solution existence to the ICP problem for any number of points.

Regarding structure, the paper stars with mathematical preliminaries on dual numbers, dual vectors and dual tensors. Next, kinematic aspects of tensorial parameterization of motion are discussed. Section IV presents the new approach for finding the ICP initial guess, while section V details some numerical results. In the end conclusions are drawn and future work is discussed.

II. Mathematical Preliminaries

In this section we will present some properties of the main algebraic sets used in our work: dual numbers, dual vectors and dual tensors. More details on dual numbers, dual vectors and dual tensors can be found in

Consider the set of real dual numbers to be denoted by

$$\underline{\mathbb{R}} = \mathbb{R} + \varepsilon\mathbb{R} = \{\underline{a} = a + \varepsilon a_0 |\ a, a_0 \in \mathbb{R},\ \varepsilon^2 = 0\}, \quad (4)$$

where $a = Re(\underline{a})$ is the real part of $\underline{a}$ and $a_0 = Du(\underline{a})$ the imaginary part. In the Euclidean space, the linear space of free vectors with dimension 3 will be denoted by V_3. The ensemble of dual vectors is defined as

$$\underline{V_3} = V_3 + \varepsilon V_3 = \{\underline{\mathbf{a}} = \mathbf{a} + \varepsilon \mathbf{a}_0; \mathbf{a}, \mathbf{a}_0 \in V_3,\ \varepsilon^2 = 0\}, \quad (5)$$

where $\mathbf{a} = Re(\underline{\mathbf{a}})$ is the real part of $\underline{\mathbf{a}}$ and $\mathbf{a}_0 = Du(\underline{\mathbf{a}})$ the imaginary part.

An Euclidean dual tensor represent an $\underline{\mathbb{R}}$-linear application of $\underline{V}_3$ into $\underline{V}_3$, where:

$$\begin{gathered}\underline{\boldsymbol{T}}(\underline{\lambda}_1 \underline{\mathbf{v}}_1 + \underline{\lambda}_2 \underline{\mathbf{v}}_2) = \underline{\lambda}_1 \underline{\boldsymbol{T}}(\underline{\mathbf{v}}_1) + \underline{\lambda}_2 \underline{\boldsymbol{T}}(\underline{\mathbf{v}}_2), \\ \forall \underline{\lambda}_1, \underline{\lambda}_2 \in \underline{\mathbb{R}}, \forall \underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2 \in \underline{\mathbf{V}}_3.\end{gathered} \quad (6)$$

From now on, any Euclidean dual tensor will be shortly called dual tensor and $\mathbf{L}(\underline{V}_3, \underline{V}_3)$ will denote the free $\underline{\mathbb{R}}$-module of dual tensors. Any dual tensor $\underline{\boldsymbol{T}} \in \mathbf{L}(\underline{V}_3, \underline{V}_3)$ can be decomposed in $\underline{\boldsymbol{T}} = T + \varepsilon T_0$, where $T, T_0 \in \mathbf{L}(V_3, V_3)$ are real tensors. The transposed dual tensor, denoted by $\underline{\boldsymbol{T}}^{\mathrm{T}}$, is defined by

$$\underline{\mathbf{v}}_1 \cdot (\underline{\boldsymbol{T}} \underline{\mathbf{v}}_2) = \underline{\mathbf{v}}_2 \cdot (\underline{\boldsymbol{T}}^{\mathrm{T}} \underline{\mathbf{v}}_1), \quad \forall \underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2 \in \underline{V}_3. \quad (7)$$

while $\forall \underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2, \underline{\mathbf{v}}_3 \in \underline{V}_3, Re < \underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2, \underline{\mathbf{v}}_3 > \neq 0$ the determinant is:

$$< \underline{\boldsymbol{T}} \underline{\mathbf{v}}_1, \underline{\boldsymbol{T}} \underline{\mathbf{v}}_2, \underline{\boldsymbol{T}} \underline{\mathbf{v}}_3 > = \det \underline{\boldsymbol{T}} < \underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2, \underline{\mathbf{v}}_3 > . \quad (8)$$

An important class of invariants that will be used to describe the dual tensor are called **linear invariants** and are denoted by $\mathrm{vect}\underline{\boldsymbol{T}} = \mathrm{vect}\frac{1}{2}[\underline{\boldsymbol{T}} - \underline{\boldsymbol{T}}^{\mathrm{T}}]$, $\mathrm{trace}\underline{\boldsymbol{T}}$ [13], where

$$\mathrm{trace}\underline{\boldsymbol{T}} = \frac{< \underline{\boldsymbol{T}}\underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2, \underline{\mathbf{v}}_3 > + < \underline{\mathbf{v}}_1, \underline{\boldsymbol{T}}\underline{\mathbf{v}}_2, \underline{\mathbf{v}}_3 > + < \underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2, \underline{\boldsymbol{T}}\underline{\mathbf{v}}_3 >}{< \underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2, \underline{\mathbf{v}}_3 >} \quad (9)$$

for any $\underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2, \underline{\mathbf{v}}_3 \in \underline{V}_3$ with $Re(< \underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2, \underline{\mathbf{v}}_3 >) \neq \mathbf{0}$.

Let the orthogonal dual tensor set be denoted by

$$\underline{S\mathbb{O}}_3 = \{\underline{\boldsymbol{R}} \in \mathbf{L}(\underline{\mathbf{V}}_3, \underline{\mathbf{V}}_3) \mid \underline{\boldsymbol{R}}\underline{\boldsymbol{R}}^T = \underline{\boldsymbol{I}},\ \det \underline{\boldsymbol{R}} = 1\}, \quad (10)$$

where $\underline{\boldsymbol{I}}$ is the unit orthogonal dual tensor. The internal structure of any orthogonal dual tensor $\underline{\boldsymbol{R}} \in \underline{S\mathbb{O}}_3$ is illustrated in a series of results which were detailed in our previous work [13].

If $\underline{\boldsymbol{R}}$ is an orthogonal dual tensor then for any dual vectors $\underline{\mathbf{a}}$ and $\underline{\mathbf{b}}$ the following two expression are valid:

$$\underline{\boldsymbol{R}}(\underline{\mathbf{a}}) \cdot \underline{\boldsymbol{R}}(\underline{\mathbf{b}}) = \underline{\mathbf{a}} \cdot \underline{\mathbf{b}}, \quad (11)$$

$$\underline{\boldsymbol{R}}(\underline{\mathbf{a}}) \times \underline{\boldsymbol{R}}(\underline{\mathbf{b}}) = \underline{\boldsymbol{R}}(\underline{\mathbf{a}} \times \underline{\mathbf{b}}). \quad (12)$$

For any $\underline{\boldsymbol{R}} \in \underline{S\mathbb{O}}_3$, a unique decomposition is viable

$$\underline{\boldsymbol{R}} = (I + \varepsilon\widetilde{\boldsymbol{\rho}})\mathbf{Q}, \quad (13)$$

where $Q \in S\mathbb{O}_3$ and $\boldsymbol{\rho} \in V_3$ are called **structural invariants** of $\underline{\boldsymbol{R}} \in \underline{S\mathbb{O}_3}$.

It can easily be proven that $\underline{S\mathbb{O}_3}$ is a Lie group. This property combined with the result presented in equation (13), leads to the conclusion that any orthogonal dual tensor $\underline{\boldsymbol{R}} \in \underline{S\mathbb{O}}_3$ can be used to globally parameterize displacements of rigid bodies. This aspect will be detailed in the following section.

III. Tensorial parameterization of motion

In this section an isomorphism between the Lie group SE_3 and the Lie group $\underline{S\mathbb{O}}_3$ is presented. This isomorphism will further be used in the development of the algorithm for finding the initial guess to the ICP problem. A series of theorems and remarks that will be used to map the ICP problem from SE_3 into $\underline{S\mathbb{O}}_3$ are detailed next:

Theorem 1. *The special Euclidean group* $(SE_3, \cdot)$ *and* $(\underline{S\mathbb{O}}_3, \cdot)$ *are connected via the isomorphism*

$$\Phi : SE_3 \to \underline{S\mathbb{O}}_3, \quad \Phi(g) = (I + \varepsilon\widetilde{\boldsymbol{\rho}})\,Q, \tag{14}$$

where $g = \begin{bmatrix} Q & \boldsymbol{\rho} \\ \mathbf{0} & 1 \end{bmatrix}$, $\widetilde{\boldsymbol{\rho}}$ *is the real antisymmetric tensor linked with the real vector* $\boldsymbol{\rho}$.

Remark 1. *The inverse of* Φ *is*

$$\Phi^{-1} : \underline{S\mathbb{O}}_3 \leftrightarrow SE_3; \quad \Phi^{-1}(\underline{\boldsymbol{R}}) = \begin{bmatrix} Q & \boldsymbol{\rho} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{15}$$

where, $Q = Re(\underline{\boldsymbol{R}})$, $\boldsymbol{\rho} = vect(Du(\underline{\boldsymbol{R}})Q^{\mathrm{T}})$.

Given two dual vectors $\underline{\mathbf{a}}$ and $\underline{\mathbf{b}} \in \underline{V}_3$, $\underline{\mathbf{a}} \otimes \underline{\mathbf{b}}$ denotes a dual tensor called ***tensor (dyadic) product*** and is defined by:

$$\underline{\mathbf{a}} \otimes \underline{\mathbf{b}} : \underline{V}_3 \to \underline{V}_3, \quad (\underline{\mathbf{a}} \otimes \underline{\mathbf{b}})\underline{\mathbf{v}} = (\underline{\mathbf{v}} \cdot \underline{\mathbf{b}})\underline{\mathbf{a}}, \ \forall \underline{\mathbf{v}} \in \underline{V}_3. \tag{16}$$

Among the properties of (16) we enumerate:

$$\text{trace } \underline{\mathbf{a}} \otimes \underline{\mathbf{b}} = \underline{\mathbf{a}} \cdot \underline{\mathbf{b}}, \tag{17}$$

$$\text{vect } \underline{\mathbf{a}} \otimes \underline{\mathbf{b}} = \frac{1}{2}\underline{\mathbf{b}} \times \underline{\mathbf{a}}, \tag{18}$$

$$(\underline{\mathbf{a}} \otimes \underline{\mathbf{b}})(\underline{\mathbf{c}} \otimes \underline{\mathbf{d}}) = (\underline{\mathbf{b}} \cdot \underline{\mathbf{c}})\underline{\mathbf{a}} \otimes \underline{\mathbf{d}}. \tag{19}$$

Remark 2. *Consider a set of dual vectors* $\{\underline{\mathbf{b}}_1, \underline{\mathbf{b}}_2, ..., \underline{\mathbf{b}}_n\} \in \underline{V}_3$. *If* $\exists\ i, j, k \in \{1, 2, ..., n\}$ *with* $Re(<\underline{\mathbf{b}}_i, \underline{\mathbf{b}}_j, \underline{\mathbf{b}}_k>) \neq 0$ *then the dual tensor*

$$\underline{\boldsymbol{S}} = \underline{\mathbf{b}}_1 \otimes \underline{\mathbf{b}}_1 + \underline{\mathbf{b}}_2 \otimes \underline{\mathbf{b}}_2 + ... + \underline{\mathbf{b}}_n \otimes \underline{\mathbf{b}}_n, \tag{20}$$

is symmetric and invertible.

Proof. The dual tensor $\underline{\boldsymbol{S}} = S + \varepsilon S_0$ is invertible if and only if $S = Re(\underline{\boldsymbol{S}})$ is invertible [15]. Taking into account the construction of $\underline{\boldsymbol{S}}$ results that its real part is $S = \mathbf{b}_1 \otimes \mathbf{b}_1 + \mathbf{b}_2 \otimes \mathbf{b}_2 + ... + \mathbf{b}_n \otimes \mathbf{b}_n$. Using the hypothesis $Re <\underline{\mathbf{b}}_i, \underline{\mathbf{b}}_j, \underline{\mathbf{b}}_k> \neq 0$ results that $<\mathbf{b}_i, \mathbf{b}_j, \mathbf{b}_k> \neq 0$ which underlines that S is invertible and thus the proof of the remark is completed. □

Remark 3. *Consider a set of dual vectors* $\{\underline{\mathbf{b}}_1, \underline{\mathbf{b}}_2, ..., \underline{\mathbf{b}}_n\} \in \underline{V}_3$. *If only two dual vectors that fulfill* $Re(\underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j) \neq \mathbf{0}$, $i, j \in \{1, 2, ..., n\}$ *are included in the set, the dual tensor* $\underline{\boldsymbol{S}}$

$$\underline{\boldsymbol{S}} = \underline{\mathbf{b}}_1 \otimes \underline{\mathbf{b}}_1 + \underline{\mathbf{b}}_2 \otimes \underline{\mathbf{b}}_2 + ... + \underline{\mathbf{b}}_n \otimes \underline{\mathbf{b}}_n + (\underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j) \otimes (\underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j) \tag{21}$$

is symmetric and invertible.

Proof. The triple scalar product of the dual vectors $\{\underline{\mathbf{b}}_i, \underline{\mathbf{b}}_j, \underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j\}$ implies $Re(<\underline{\mathbf{b}}_i, \underline{\mathbf{b}}_j, \underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j> = Re(|\underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j|^2) \neq 0)$. This allows the application of Remark 2 over the dual vectors set $\{\underline{\mathbf{b}}_1, \underline{\mathbf{b}}_2, ..., \underline{\mathbf{b}}_n, \underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j\}$, thus finalizing our proof. □

Using the previous remarks we can define the **reciprocal dual vectors** of $\{\underline{\mathbf{b}}_1, \underline{\mathbf{b}}_2, ..., \underline{\mathbf{b}}_n\}$, $\underline{\mathbf{b}}_i \in \underline{V}_3$, $i = \overline{1, n}$:

$$\underline{\mathbf{b}}_i^* = \underline{\boldsymbol{S}}^{-1}\underline{\mathbf{b}}_i, i = \overline{1, n}. \tag{22}$$

Using this definition we can easily prove that

$$\underline{\mathbf{b}}_1 \otimes \underline{\mathbf{b}}_1^* + \underline{\mathbf{b}}_2 \otimes \underline{\mathbf{b}}_2^* + ... + \underline{\mathbf{b}}_n \otimes \underline{\mathbf{b}}_n^* = \underline{\boldsymbol{I}} \tag{23}$$

due to the transpose of:

$$\sum_{i=1}^{n} \underline{\mathbf{b}}_i^* \otimes \underline{\mathbf{b}}_i = \sum_{i=1}^{n} [\underline{\boldsymbol{S}}^{-1}\underline{\mathbf{b}}_i] \otimes \underline{\mathbf{b}}_i = \underline{\boldsymbol{S}}^{-1} \sum_{i=1}^{n} \underline{\mathbf{b}}_i \otimes \underline{\mathbf{b}}_i = \underline{\boldsymbol{S}}^{-1}\underline{\boldsymbol{S}} = \underline{\boldsymbol{I}}. \tag{24}$$

This observation allows that any dual vector $\underline{\mathbf{v}} \in \underline{V}_3$ to be decomposed into

$$\underline{\mathbf{v}} = \underline{\alpha}_1 \underline{\mathbf{b}}_1 + \underline{\alpha}_2 \underline{\mathbf{b}}_2 + ... + \underline{\alpha}_n \underline{\mathbf{b}}_n \tag{25}$$

where $\underline{\alpha}_i = \underline{\mathbf{v}} \cdot \underline{\mathbf{b}}_i^*, \ i = \overline{1, n}$.

Theorem 2. *Consider a set of dual vectors* $\{\underline{\mathbf{b}}_1, \underline{\mathbf{b}}_2, ..., \underline{\mathbf{b}}_n\}, \underline{\mathbf{b}}_i \in \underline{V}_3, \ i = \overline{1, n}, \ n > 3$ *that includes at least two dual vectors that fulfill* $Re(\underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j) \neq 0$. *For any set of dual vectors* $\{\underline{\mathbf{a}}_1, \underline{\mathbf{a}}_2, ..., \underline{\mathbf{a}}_n, \}, \underline{\mathbf{a}}_i \in \underline{V}_3, \ i = \overline{1, n}$,*the following two results are true:*

(i) if a unique dual tensor $\underline{\boldsymbol{T}} \in L(\underline{V}_3, \underline{V}_3)$ *exists in order to have:* $\underline{\boldsymbol{T}}\underline{\mathbf{b}}_i = \underline{\mathbf{a}}_i, \ \ i = \overline{1, n+1}$, *the dual tensor can be computed using:*

$$\underline{\boldsymbol{T}} = \underline{\mathbf{a}}_1 \otimes \underline{\mathbf{b}}_1^* + \underline{\mathbf{a}}_2 \otimes \underline{\mathbf{b}}_2^* + ... + \underline{\mathbf{a}}_n \otimes \underline{\mathbf{b}}_n^* + (\underline{\mathbf{a}}_i \times \underline{\mathbf{a}}_j) \otimes (\underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j)^* \tag{26}$$

(ii)

$$\underline{\boldsymbol{T}} \in \underline{S\mathbb{O}}_3 \Leftrightarrow \underline{\mathbf{a}}_i \cdot \underline{\mathbf{a}}_j = \underline{\mathbf{b}}_i \cdot \underline{\mathbf{b}}_j, \ i, j = \overline{1, n+1} \tag{27}$$

We have denoted by $\underline{\mathbf{a}}_{n+1}$, $\underline{\mathbf{b}}_{n+1}$ *the followings:* $\underline{\mathbf{a}}_{n+1} = \underline{\mathbf{a}}_i \times \underline{\mathbf{a}}_j$ *and* $\underline{\mathbf{b}}_{n+1} = \underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j$.

Theorem 3. *The system of equations*

$$\underline{\boldsymbol{R}}\underline{\mathbf{b}}_i = \underline{\mathbf{a}}_i, \ i = \overline{1, n}, \ n \in \mathbb{N}^*, \ n > 3 \tag{28}$$

has a unique solution $\underline{\boldsymbol{R}} \in \underline{S\mathbb{O}}_3$ *if and only if*

$$\begin{cases} \underline{\mathbf{a}}_i \cdot \underline{\mathbf{a}}_j = \underline{\mathbf{b}}_i \cdot \underline{\mathbf{b}}_j, \quad i, j = \overline{1, n+1} \\ \exists i, j \in \overline{1, n} \text{ in order to have } Re(\underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j) \neq \mathbf{0} \end{cases} \tag{29}$$

We have denoted by $\underline{\mathbf{a}}_{n+1}$, $\underline{\mathbf{b}}_{n+1}$ *the followings:* $\underline{\mathbf{a}}_{n+1} = \underline{\mathbf{a}}_i \times \underline{\mathbf{a}}_j$ *and* $\underline{\mathbf{b}}_{n+1} = \underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j$.

If $\exists i, j, k \in \{1, 2, ..., n\}$ *with* $<\underline{\mathbf{b}}_i, \underline{\mathbf{b}}_j, \underline{\mathbf{b}}_k> = <\underline{\mathbf{a}}_i, \underline{\mathbf{a}}_j, \underline{\mathbf{a}}_k>$, *and* $Re(<\underline{\mathbf{b}}_i, \underline{\mathbf{b}}_j, \underline{\mathbf{b}}_k>) \neq 0$, *the solution is*

$$\underline{\boldsymbol{R}} = \underline{\mathbf{a}}_1 \otimes \underline{\mathbf{b}}_1^* + \underline{\mathbf{a}}_2 \otimes \underline{\mathbf{b}}_2^* + ... + \underline{\mathbf{a}}_n \otimes \underline{\mathbf{b}}_n^*, \tag{30}$$

where $\underline{\mathbf{b}}_i^*, i = \overline{1, n}$ *are the reciprocal vectors of* $\underline{\mathbf{b}}_i, i = \overline{1, n}$. *Else the solution is:*

$$\underline{\boldsymbol{R}} = \underline{\mathbf{a}}_1 \otimes \underline{\mathbf{b}}_1^* + \underline{\mathbf{a}}_2 \otimes \underline{\mathbf{b}}_2^* + ... + \underline{\mathbf{a}}_n \otimes \underline{\mathbf{b}}_n^* + \underline{\mathbf{a}}_{n+1} \otimes \underline{\mathbf{b}}_{n+1}^*. \tag{31}$$

Now, the layout is set for the development of the new method of computing the initial guess to ICP problem.

IV. A new method for computing the ICP Initial Guess

A complete closed-form solution to the registration problem is presented next. Also, a new method for computing the ICP initial guess is detailed and a computational procedure is illustrated.

A. Point-to-Point Registration

Consider a set of points A that needs to be register to another set of points B, where the number of elements of either A or B is n. If the point-to-point correspondents are known, we can provide an algebraic closed form solution. Consider $\mathbf{c}_A$ and $\mathbf{c}_B$ to be the position vectors for the centroids of the two sets. For each point $\mathbf{a}_i \in A$ and $\mathbf{b}_i \in B$ the following dual vectors can be computed:

$$\begin{aligned} \underline{\mathbf{a}}_i &= \mathbf{a}_i - \mathbf{c}_A + \varepsilon \mathbf{c}_A \times \mathbf{a}_i, \quad i = \overline{1,n} \\ \underline{\mathbf{b}}_i &= \mathbf{b}_i - \mathbf{c}_B + \varepsilon \mathbf{c}_B \times \mathbf{b}_i, \quad i = \overline{1,n} \end{aligned} . \tag{32}$$

For a set of dual vectors computed as in (32), the system

$$\underline{\boldsymbol{X}}\underline{\mathbf{b}}_i = \underline{\mathbf{a}}_i, \quad \underline{\mathbf{a}}_i, \underline{\mathbf{b}}_i \in \underline{V}_3, \quad i = \overline{1,n}$$

has a unique solution $\underline{\boldsymbol{X}} \in \underline{S\mathbb{O}}_3$ if and only if

$$\begin{cases} \underline{\mathbf{a}}_i \cdot \underline{\mathbf{a}}_j = \underline{\mathbf{b}}_i \cdot \underline{\mathbf{b}}_j, \quad i,j = \overline{1,n+1} \\ \exists i,j \in \overline{1,n} \text{ in order to have } Re(\underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j) \neq \mathbf{0} \end{cases} . \tag{33}$$

We have denoted by $\underline{\mathbf{a}}_{n+1}$, $\underline{\mathbf{b}}_{n+1}$ the followings: $\underline{\mathbf{a}}_{n+1} = \underline{\mathbf{a}}_i \times \underline{\mathbf{a}}_j$ and $\underline{\mathbf{b}}_{n+1} = \underline{\mathbf{b}}_i \times \underline{\mathbf{b}}_j$.

If $\exists i,j,k \in \{1,2,...,n\}$ with $< \underline{\mathbf{b}}_i, \underline{\mathbf{b}}_j, \underline{\mathbf{b}}_k > = < \underline{\mathbf{a}}_i, \underline{\mathbf{a}}_j, \underline{\mathbf{a}}_k >$ and $Re(< \underline{\mathbf{b}}_i, \underline{\mathbf{b}}_j, \underline{\mathbf{b}}_k >) \neq 0$ the solution is

$$\underline{\boldsymbol{X}} = \sum_{i=1}^{n} \underline{\mathbf{a}}_i \otimes \underline{\mathbf{b}}_i^* \tag{34}$$

else the solution is

$$\underline{\boldsymbol{X}} = \sum_{i=1}^{n+1} \underline{\mathbf{a}}_i \otimes \underline{\mathbf{b}}_i^* . \tag{35}$$

The last results solves the problem of finding the closed-form simultaneous solution to the general point-to-point registration problem.

B. Initial guess for the ICP Problem

In the previous subsection it was detailed how the point-to-point registration problem can be directly solved through an algebraic approach. Taking into account that usually the full correspondence between the two point sets is not available, the method we propose for computing the initial guess of the ICP problem is related to finding the closest orthogonal tensor to noisy data of point-to-point registration problem.

Two filtering techniques are presented next, together with the algorithm that can be used to put into practice the solution to the point-to-point registration problem when disturbances are considered.

- **QR** filtering procedure

Algorithm 1: Computational procedure for finding the initial guess of the ICP problem

Input data $\mathbf{a}_i, \mathbf{b}_i \in \mathbb{R}^3, i = \overline{1,n}$
Compute the position vectors for the centroids of both point sets
Compute $\underline{\mathbf{a}}_i$ and $\underline{\mathbf{b}}_i$ using equations (32).
For $m \geq 2$, m being the number of the points correspondences, we need to follow the next steps:
Compute the dual tensor $\underline{\boldsymbol{S}}$ according to (20)
Compute the reciprocal dual vectors $\underline{\mathbf{b}}_i^* = \underline{\boldsymbol{S}}^{-1}\underline{\mathbf{b}}_i, \quad i = \overline{1,n}$, using equation (22).
Compute the initial guess to the ICP problem using equations (34) or (35).
Apply one the two filtering procedures (QR or SVD) to recover $\underline{\boldsymbol{X}}^{\dagger}$ or $\underline{\boldsymbol{X}}^{\dagger\dagger}$
Output data The initial guess $\mathbf{X}^{\dagger} = \Phi^{-1}(\underline{\boldsymbol{X}}^{\dagger}), \mathbf{X}^{\dagger} \in SE_3$ or $\mathbf{X}^{\dagger\dagger} = \Phi^{-1}(\underline{\boldsymbol{X}}^{\dagger\dagger}), \mathbf{X}^{\dagger\dagger} \in SE_3$.

The following filtering procedure is an adjustment over the classical Gram-Schmidt algorithm. The result of the filtering procedure is similar to the QR decomposition used in [15], [16]. Consider $\underline{\mathbf{c}}_i,\ i = \overline{1,3}$, to be the columns of the dual tensor $\underline{\boldsymbol{R}}$ which was built from noisy measurements. The correction procedure performs the Gram-Schmidt algorithm but does not divide any dual vector by its length until the end of the algorithm. This approach allows the algorithm to avoid multiple problems that can emerge when multiplying a dual vector with the inverse of a dual number.

The procedure starts by setting:

$$\underline{\mathbf{v}}_1 = \underline{\mathbf{c}}_1. \tag{36}$$

Next $\underline{\mathbf{v}}_2$ is computed from

$$\underline{\mathbf{v}}_2 = \underline{\mathbf{c}}_2 - \left(\frac{\underline{\mathbf{c}}_2 \cdot \underline{\mathbf{v}}_1}{\underline{\mathbf{v}}_1 \cdot \underline{\mathbf{v}}_1}\right)\underline{\mathbf{v}}_1, \tag{37}$$

while $\underline{\mathbf{v}}_3$ is generated by:

$$\underline{\mathbf{v}}_3 = \underline{\mathbf{c}}_3 - \left(\frac{\underline{\mathbf{c}}_3 \cdot \underline{\mathbf{v}}_1}{\underline{\mathbf{v}}_1 \cdot \underline{\mathbf{v}}_1}\right)\underline{\mathbf{v}}_1 - \left(\frac{\underline{\mathbf{c}}_3 \cdot \underline{\mathbf{v}}_2}{\underline{\mathbf{v}}_2 \cdot \underline{\mathbf{v}}_2}\right)\underline{\mathbf{v}}_2. \tag{38}$$

The resulting dual vectors $\underline{\mathbf{v}}_i,\ i = \overline{1,3}$ are orthogonal but their magnitude is not equal to $1 + \varepsilon 0$ so the final step is:

$$\underline{\mathbf{c}}_i^{\dagger} = \frac{\underline{\mathbf{v}}_i}{|\underline{\mathbf{v}}_i|},\ i = \overline{1,3}. \tag{39}$$

In the end we recover $\underline{\mathbf{R}}^{\dagger} = [\underline{\mathbf{c}}_1 \ \ \underline{\mathbf{c}}_2 \ \ \underline{\mathbf{c}}_3] \in \underline{S\mathbb{O}}_3$.

- **SVD** filtering procedure

Let $\underline{\mathbf{R}} = [\underline{\mathbf{c}}_1\ \underline{\mathbf{c}}_2\ \underline{\mathbf{c}}_3]$ be the dual matrix attached to the dual tensor $\underline{\boldsymbol{R}}$. This dual matrix has a series of properties which can be used to filter noisy data. Its singular value decomposition (SVD) should be [15]

$$\underline{\mathbf{R}} = \underline{\mathbf{U}}\ \underline{\Sigma}\ \underline{\mathbf{V}}^T, \tag{40}$$

where $\underline{\Sigma} = diag\{\underline{\sigma}_1, \underline{\sigma}_2, \underline{\sigma}_3\}$, and $\underline{\mathbf{U}}, \underline{\mathbf{V}} \in \underline{\mathbb{O}}_3$, $\underline{\mathbf{U}}\underline{\mathbf{U}}^{\mathrm{T}} = \underline{\mathbf{I}}$, $\underline{\mathbf{V}}\underline{\mathbf{V}}^{\mathrm{T}} = \underline{\mathbf{I}}$.

After SVD is performed, the orthogonal dual matrix that will be used to carry out future experiments is build using:

$$\underline{\mathbf{R}}^{\dagger\dagger} = \ \underline{\mathbf{U}}\ diag\{1,\ 1, \det\underline{\mathbf{U}} \cdot \det\underline{\mathbf{V}}\}\ \ \underline{\mathbf{V}}^{\mathrm{T}}, \quad \underline{\mathbf{R}}^{\dagger\dagger} \in \underline{S\mathbb{O}}_3 \tag{41}$$

All of the previous results can be summarized in Algorithm 1, which can easily be used to put into practice the proposed method for finding the initial guess of the ICP problem.

V. Numerical results

LIBICP (LIBrary for Iterative Closest Point fitting) [17] is a cross-platfrom C++ library with MATLAB wrappers for fitting 2d or 3d point clouds with respect to each other. Currently it implements the SVD-based point-to-point algorithm as well as the linearized point-to-plane algorithm. It also supports outliers rejection and is accelerated by the use of k-d trees as well as a coarse matching stage using only a subset of all points.

The algebraic framework described in section III was implemented in Matlab, which allowed for Algorithm 1 to be easily used together with LIBICP. The testing procedure implying the following steps:

- generate two point sets, while retaining the ground truth transformation, denoted by $\mathbf{G}$;
- compute the solution to the ICP problem with out using initial guess, denoted by $\mathbf{T}_r$;
- compute the initial guess using Algorithm 1;
- compute the solution to the ICP problem with the initial guess generated by Algorithm 1, denoted by $\mathbf{T}_{r\mathbf{X}}$.

Different test cases were analyzed, each revealing how much the existence of the initial guess influences the final result. The chosen point sets contain a number of 289 elements. The ground truth transformation was keep the same for all the cases of study, it's value being:

$$\mathbf{G} = \begin{bmatrix} 1.0000 & 0.0000 & -0.0000 & -2.0000 \\ -0.0000 & 0.9553 & -0.2955 & 1.0000 \\ 0.0000 & 0.2955 & 0.9553 & -0.0000 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \tag{42}$$

The first test considered two point sets with reduced planarity, as depicted in Fig. 1. For this configuration both $\mathbf{T}_r$ and $\mathbf{T}_{r\mathbf{X}}$ are equal to $\mathbf{G}$

As we increase the planarity of the two point sets the results show a bias between $\mathbf{T}_r$ and $\mathbf{T}_{r\mathbf{X}}$. The second test, which is depicted in Fig. 2, generated the following numerical results:

$$\mathbf{T}_r = \begin{bmatrix} 1.0000 & -0.0022 & 0.0071 & -1.8131 \\ 0.0042 & 0.9560 & -0.2934 & 0.7886 \\ -0.0061 & 0.2934 & 0.9560 & -0.0864 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \tag{43}$$

$$\mathbf{T}_{r\mathbf{X}} = \begin{bmatrix} 1.0000 & 0.0000 & -0.0000 & -2.0000 \\ -0.0000 & 0.9553 & -0.2955 & 1.0000 \\ 0.0000 & 0.2955 & 0.9553 & -0.0000 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \tag{44}$$

These results show that the solution generated without an initial guess is biased both on orientation and on translation in comparison with the ground thrush. For this case the result obtained with the initial guess is equal to the ground truth.

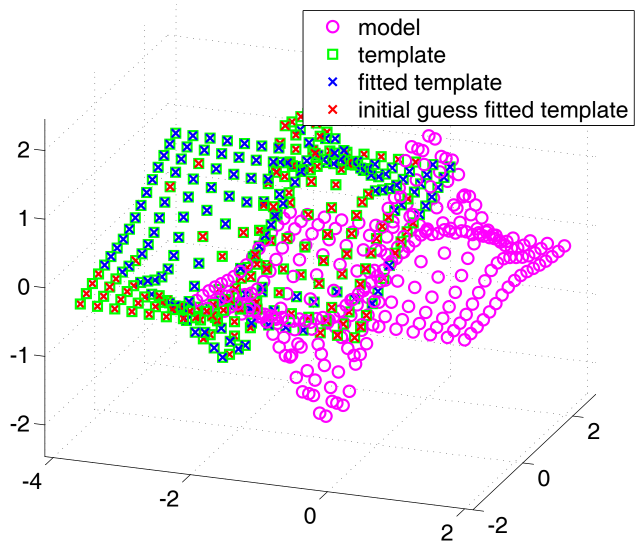

Fig. 1. First test

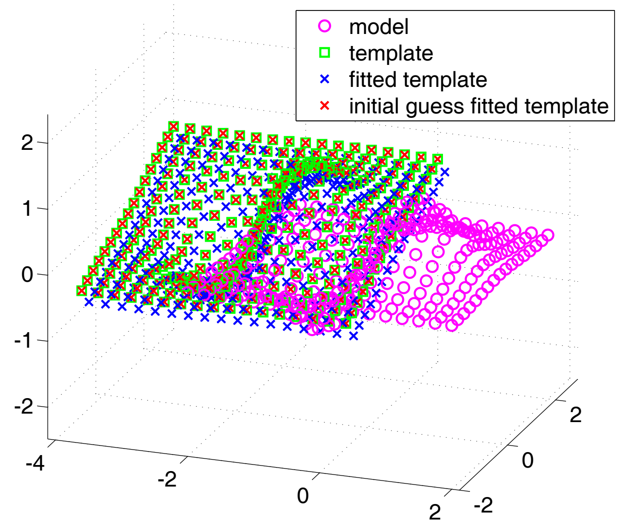

Fig. 2. Second test

Last test considered both planarity and some normal distributed disturbances. The considered disturbances had a mean equal to 0.1 and a variance of 0.01. The third test, which is depicted in Fig. 3, generated the following numerical results:

$$\mathbf{T}_r = \begin{bmatrix} 1.0000 & 0.0000 & 0.0007 & -1.4454 \\ 0.0002 & 0.9549 & -0.2969 & 0.4684 \\ -0.0007 & 0.2969 & 0.9549 & -0.0937 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \tag{45}$$

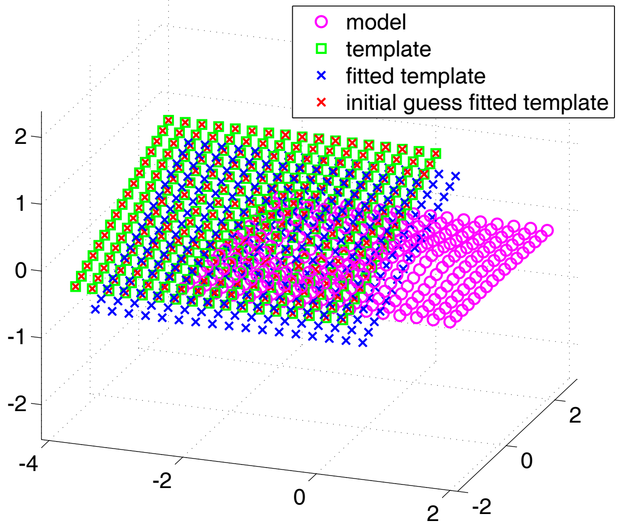

Fig. 3. Third test

$$\mathbf{X} = \begin{bmatrix} 0.9996 & -0.0003 & 0.0066 & -1.8861 \\ 0.0005 & 0.9547 & -0.3026 & 1.0927 \\ 0.0008 & 0.2953 & 0.9513 & 0.1058 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \tag{46}$$

$$\mathbf{T}_r\mathbf{x} = \begin{bmatrix} 0.9996 & -0.0003 & 0.0066 & -1.9004 \\ 0.0005 & 0.9547 & -0.3026 & 1.1008 \\ 0.0008 & 0.2953 & 0.9513 & 0.0990 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \tag{47}$$

The numerical results depicted by (45), (46) and (47), underlines the accuracy of the ICP problem solution when a robust initial guess is considered. This shows the usefulness of the proposed method, combined with the remark that tensorial approach has the capability of generating the initial guess very close to the ground truth. This will be exploited in future work when different applications, one being motion estimation, will be considered.

VI. Conclusions

This paper presents a new method for finding the initial guess to the Iterative Closest Point problem. The proposed method is based on the motion parameterization properties of orthogonal dual tensors. Using an isomorphism between the special euclidean group SE_3 and the orthogonal dual tensors group $\underline{S\mathbb{O}}_3$, a set of conditions are drawn in order to ensure the existence of closed-form simultaneous solutions for data that contains any number of elements for two set of points that need to be register. The solutions are free of coordinates and can easily be put into practice through a step by step implementation algorithm. Numerical simulations show that our solutions are accurate and robust even when noisy data are considered or singularity situations are analyzed.

Future work will include extended performance evaluation of the proposed solution when real data from sensors such as stereo vision or RGB-D.

Acknowledgements

This project has received partial funding from the European Unions Horizon 2020 research and innovation program under grant agreement No 643636 "Sound of Vision".

References

[1] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, vol. 3. IEEE, 1991, pp. 2724–2729.

[2] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[3] Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *International Journal of Computer Vision*, vol. 13, no. 2, pp. 119–152, 1994.

[4] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in *Proceedings of the Third International Conference on 3-D Digital Imaging and Modeling*, 2001.

[5] D. Chetverikov, D. Svirko, and D. Stepanov, "The trimmed iterative closest point algorithm," in *Proceedings of the 16th International Conference on Pattern Recognition*, vol. 3. IEEE, 2002, pp. 545–548.

[6] J. L. Martínez, J. González, J. Morales, A. Mandow, and A. J. GarcíaCerezo, "Mobile robot motion estimation by 2d scan matching with genetic and iterative closest point algorithms," *Journal of Field Robotics*, vol. 23, no. 1, pp. 21–34, 2006.

[7] S. Du, N. Zheng, S. Ying, and J. Liu, "Affine iterative closest point algorithm for point set registration," *Pattern Recognition Letters*, vol. 31, no. 9, pp. 791–799, 2010.

[8] S. Du, N. Zheng, L. Xiong, S. Ying, and J. Xue, "Scaling iterative closest point algorithm for registration of md point sets," *Journal of Visual Communication and Image Representation*, vol. 21, no. 5-6, pp. 442–452, 2010.

[9] S. Du, J. Liu, C. Zhang, J. Zhu, and K. Li, "Probability iterative closest point algorithm for m-d point set registration with noise," *Neurocomputing*, vol. 157, pp. 187–198, 2015.

[10] S. Du, J. Liu, B. Bi, J. Zhu, and J. Xue, "New iterative closest point algorithm for isotropic scaling registration of point sets with noise," *Journal of Visual Communication and Image Representation*, vol. 38, no. C, pp. 207–216, 2016.

[11] C. Zhang, S. Du, J. Liu, and J. Xue, "Robust 3d point set registration using iterative closest point algorithm with bounded rotation angle," *Signal Processing*, vol. 120, no. C, pp. 777–788, 2015.

[12] C. Zhang, S. Du, J. Liu, Y. Li, J. Xue, and Y. Liu, "Robust iterative closest point algorithm with bounded rotation angle for 2d registration," *Neurocomputing*, vol. 195, no. C, pp. 172–180, 2016.

[13] D. Condurache and A. Burlacu, "Dual tensors based solutions for rigid body motion parameterization," *Mechanism and Machine Theory*, vol. 74, pp. 390–412, 2014.

[14] ——, "Orthogonal dual tensor method for solving the $\mathbf{AX} = \mathbf{XB}$ sensor calibration problem," *Mechanism and Machine Theory*, vol. 104, pp. 382–404, 2016.

[15] E. Pennestri and P. Valentini, "Linear dual algebra algorithms and their application to kinematics," *Multibody Dynamics: Computational Methods and Applications*, vol. 12, pp. 207–229, 2009.

[16] J. Angeles, "The dual generalized inverses and their applications in kinematic synthesis," in *Latest Advances in Robot Kinematics*, J. Lenarcic and M. Husty, Eds. Springer International Publishing, 2012, pp. 1–10.

[17] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.