



Estimating Rigid Transformations of Noisy Point Clouds Using the Universal Manifold Embedding

Amit Efraim¹ · Joseph M. Francos¹

Received: 15 June 2021 / Accepted: 20 January 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

We present a closed form solution to the problem of registration of fully overlapping 3D point clouds undergoing unknown rigid transformations, as well as for detection and registration of sub-parts undergoing unknown rigid transformations. The solution is obtained by adapting the general framework of the universal manifold embedding (UME) to the case where the transformations the object may undergo are rigid. The UME nonlinearly maps functions related by certain types of geometric transformations of coordinates to the same linear subspace of some Euclidean space while retaining the information required to recover the transformation. Therefore registration, matching and classification can be solved as linear problems in a low dimensional linear space. In this paper, we extend the UME framework to the special case where it is *a priori* known that the geometric transformations are rigid. While a variety of methods exist for point cloud registration, the method proposed in this paper is notably different as registration is achieved by a closed form solution that employs the UME low dimensional representation of the shapes to be registered.

Keywords Rigid transformations · Registration · Parameter Estimation · Deformable templates · Point clouds

1 Introduction

Registration of 3D point clouds has been an active research subject with a vast range of applications from problems in computer vision, robotics and autonomous navigation, to medical imaging. There are many approaches to the 3D point cloud registration problem. One of the most commonly practiced approaches is to extract and match spatially local features (*e.g.*, [25,31,47,55,57,58]).

Many of the existing methods are adaptations to 3D of image processing solutions, such as variants of 3D-SIFT [15,35] and the Harris keypoint detector [48]. In 3D, with the absence of a regular sampling grid, artifacts, sampling

noise and the challenging nature of salient geometry (edges are not common in 3D as in images, for example), keypoint matching is prone to high outlier rates and localization errors. Hence, the alignment estimated by keypoint matching usually employs outlier rejection methods such as RANSAC [18] and Guaranteed Outliers Rejection [8], or directly solve an optimization problem such that outliers are discarded directly during optimization as in Fast Global Registration [62] and TEASER [54]. These registration algorithms are usually followed by a refinement stage using local optimization algorithms [6,36,44,61].

Refinement algorithms employ numerical optimization to iteratively minimize an objective function measuring the distance between points in the observation and assumed correspondence points in the reference model [6,61], or between points in the observation and the surface of the model [6,36,44].

The Iterative Closest Point algorithm (ICP) [6,61] is the standard algorithm in this category. It constructs point correspondences based on spatial proximity followed by a transformation estimation step. Over the years, many variants of the ICP algorithm have been proposed in attempt to improve the convergence rate, robustness and accuracy of the algorithm. In [10] for example, not all points in both models

This research is supported by NSF-BSF Computing and Communication Foundations (CCF) grants, CCF-2016667 and BSF-2016667.

✉ Amit Efraim
efraimam@post.bgu.ac.il

Joseph M. Francos
francos@ee.bgu.ac.il

¹ Department of Electrical and Computer Engineering, Ben-Gurion University, Beersheba, Israel

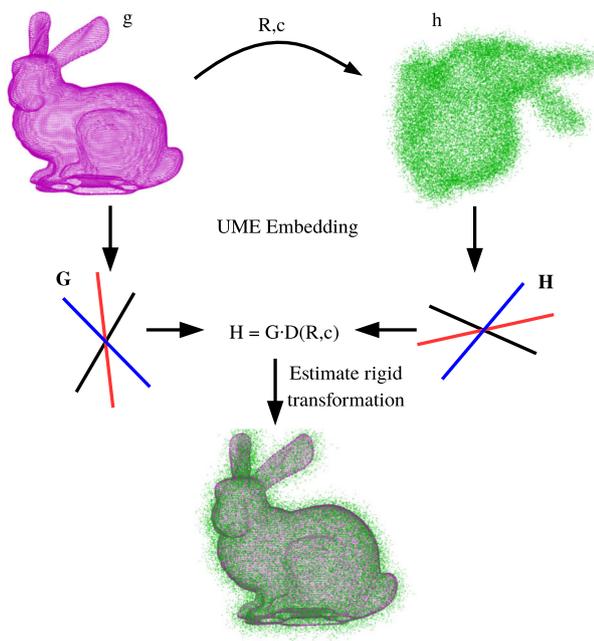


Fig. 1 Registration of a noisy Stanford Bunny model undergoing a rigid transformation (parameterized by rotation matrix \mathbf{R} and translation vector \mathbf{c}) to a noise free reference, using the Rigid Transformation Universal Manifold Embedding (RT-UME). First, models are embedded in a low dimensional linear space using (6) resulting in UME matrices \mathbf{H} and \mathbf{G} . Since the embedding is covariant with the transformation, the rigid transformation is estimated in the UME space and applied to the noisy point cloud for registration

are taken into account, and the number of obtained matches is trimmed according to an assumed overlap between the models undergoing registration. This approach overcomes bias in the registration that is due to the closest point paradigm failure (*i.e.*, when there is partial overlap the closest point may be in fact very far). Other variants of the ICP algorithm attempt to improve the formation of correspondences by including additional information such as surface normals, or varying the direction of closest point search, etc. (see [45] for a review of ICP variants). By definition, the ICP like any other iterative numerical optimization method (*e.g.*, [36,44,50,60]) requires a good initial alignment, otherwise registration may converge to a local minimum of the objective function. In addition, as presented in [45], not all variants are appropriate to all data types, and the algorithm should be matched to the data type for optimal performance.

Registration methods are not restricted only to methods based on the extraction and matching of keypoints. In [37], for example, an initial alignment is found by employing a matched filter in the frequency space of local orientation histograms [28]. In [17], an initial alignment is found by clustering the orientations of local point cloud descriptors followed by estimating the relative rotation between clusters. In [1] and [38], the algorithm searches for congruent sets of

four co-planar points between point clouds to create point correspondences. In [7], a registration procedure based on Fourier–Mellin transform is derived. It is a three step procedure where an $SO(3)$ Fourier transform implemented using spherical harmonics is employed to estimate the rotation. In the second step using the Mellin transform, the scale is estimated, and finally the translation. A different approach is to approximate the surface using Gaussian Mixture Models (GMM) and perform registration on the GMM models rather than directly on the point clouds, *e.g.*, [9,14,39] and the recent learning based DeepGMR [59]. The GMM modeling methods tend to be computationally expensive as each point in the point cloud is assumed to be the center of a model component. Another drawback is that the final result depends on the model initialization which is usually random.

More recently, deep learning algorithms have employed the PointNet architecture [20] as a building block for 3D registration [3,33,52]. [33] is a scene flow solution which by definition assumes a small movement between point clouds. Similarly, [3] is an adaptation of the Lucas & Kanade image registration technique [34] to 3D using the PointNet architecture. [52] adopts the standard approach of creating point correspondences by training a differentiable soft matching function instead of the usual “hard” matching. While being more robust to the initial alignment than ICP, these methods are applicable only when the transformation is of a small magnitude. Other learning methods employ different definitions of 3D convolutions to extract and match features for creating point correspondences, [5,13,22], followed by standard outlier rejection and registration procedures. In [12], putative matches are treated as six-dimensional points, and the correspondences are estimated by directly processing the six-dimensional geometry while estimating the probability of each putative match to be an inlier.

A different family of registration algorithms is plane-based registration where planes are detected and matched between point clouds [23,32,43,51]. Such methods, however, generally assume a small transformation between scans (*e.g.*, successive scans from a moving LIDAR scanner).

The method proposed in this paper is notably different as registration is performed using a global closed form solution that employs the UME low dimensional representation of the shapes to be registered. As a result, an efficient and accurate registration scheme is achieved where no initial alignment is required.

More specifically, in this paper we show that enforcing a rigid deformation constraint on the Universal Manifold Embedding (UME) [27], derived for the case where the geometric transformation is affine, results in an accurate and computationally efficient closed form solution to the problem of registration of fully overlapping 3D point clouds undergoing unknown rigid transformations, as well as for detection and registration of sub-parts undergoing unknown

rigid transformations. It is also shown that the suggested formulation enables simple perturbation analysis of the estimator performance in the presence of noise. Such an analysis enables a simple derivation of an unbiased estimator of the transformation parameters. The presented method is *independent* of the initial pose of the point clouds, and robust to varying data sampling rates. Since the UME low dimensional representation is also independent of the data dimensions, it can handle large data sets with linear computational complexity, which is a major limitation for existing methods.

Since the UME (and hence the rigid transformation UME) is an operator defined on functions of the coordinates, in order to enable registration, these functions need to be invariant to the transformation. Hence, as a pre-processing step each point in the point cloud is assigned a value by defining an SE(3)-invariant “coloring” function on the point cloud. There are different methodologies for defining an SE(3)-invariant coloring function on a point cloud. In the examples discussed in the following sections, two specific choices of an SE(3)-invariant coloring function are evaluated: The local curvature estimated using the volume curvature descriptor, [21], and the signed distance of the point from the plane passing through the point cloud center of mass, such that the plane normal is the axis of smallest variation of the point cloud.

We demonstrate the performance of the proposed method using examples of 3D topographic point-cloud registration through extensive numerical evaluation and comparison with state-of-the-art methods. In the evaluation, we consider the case of fully overlapping point clouds, where their relative pose is unknown, as well as an illustrative example of sub-part detection and registration where one point cloud is fully contained in the other. In the latter, an observed topographic model acquired by LIDAR or using multiview 3D reconstruction techniques is to be registered to a reference topographic model (for example, in the form of a DTM) of a larger area where it is assumed to be completely contained in the reference model. The presented method is shown to outperform the results achieved by state-of-the-art algorithms in both setups.

1.1 Contribution

The contribution of this paper is twofold. First, we establish the framework for utilizing the Universal Manifold Embedding [27] when processing noisy 3D point clouds, providing a constrained closed form solution to the problem of registering 3D point clouds related by a rigid transformation. Second, the proposed method is shown to provide an accurate and robust building block for point cloud registration methods.

1.2 Organization

The rest of this paper is structured as follows. In Sect. 2, brief technical background required for the derivation of the proposed solution is provided. In Sect. 3, the derivation of the UME for the special case of a rigid transformation is presented. In Sect. 4, model mismatch is analyzed and an unbiased estimator is derived. In Sect. 5, the Rigid Transformation UME, derived in Sect. 3, is adapted to the case where the objects undergoing rigid transformation are point clouds. In Sect. 6, experimental results are provided, and in Sect. 7, we provide our conclusions.

2 Background

Next, we briefly review the two methodologies that form the basis of the framework presented in this paper. These are the UME [27] and the closed form solution for registration of point clouds from known correspondences [4,30].

2.1 Closed Form Solution for Registration of Point Clouds from Known Correspondences

In [29,30], a closed form solution for the problem of recovering the transformation between two sets of points in different Cartesian coordinate systems is derived. More specifically, the problem is defined by

$$\mathbf{r}_{r,i} = s\mathbf{R}\mathbf{r}_{l,i} + \mathbf{r}_0, \quad i = 1, \dots, n \quad (1)$$

where $\{\mathbf{r}_{l,i}\}_{i=1}^n$ is the “left hand” (or reference) set of points mapped to the “right hand” (or observed) set of points $\{\mathbf{r}_{r,i}\}_{i=1}^n$ by a transformation defined by s , \mathbf{R} and \mathbf{r}_0 . The goal then is to find \mathbf{R} , \mathbf{r}_0 and s such that we minimize

$$\sum_{i=1}^n \|\mathbf{e}_i\|^2 = \sum_{i=1}^n \|\mathbf{r}_{r,i} - s\mathbf{R}\mathbf{r}_{l,i} - \mathbf{r}_0\|^2 \quad (2)$$

In [30], it is shown that the translation \mathbf{r}_0 that minimizes the sum of squared errors is the difference between the centroid of $\{\mathbf{r}_{r,i}\}_{i=1}^n$ and the rotated centroid of $\{\mathbf{r}_{l,i}\}_{i=1}^n$, *i.e.*, $\mathbf{r}_0 = \frac{1}{n} \sum_{i=1}^n \mathbf{r}_{r,i} - \frac{1}{n} s\mathbf{R} \sum_{i=1}^n \mathbf{r}_{l,i}$. The rotation that minimizes the sum of squared errors is shown to be the matrix maximizing $\text{tr}\{\mathbf{R}^T \mathbf{M}\}$, where $\mathbf{M} = \sum_{i=1}^n \mathbf{r}'_{r,i} \mathbf{r}'_{l,i}{}^T$ (here, $\mathbf{r}'_{r,i}$, $\mathbf{r}'_{l,i}$ represent the centered versions of $\mathbf{r}_{r,i}$ and $\mathbf{r}_{l,i}$). \mathbf{M} is shown to have a decomposition $\mathbf{M} = \mathbf{U}\mathbf{S}$ where $\mathbf{U} = \mathbf{M}(\mathbf{M}^T \mathbf{M})^{-1/2}$ is a unitary matrix and \mathbf{S} is a semi-positive definite matrix. It is then shown by analyzing the eigenvalues of the matrix \mathbf{M} that $\mathbf{R} = \mathbf{U}$ is the matrix minimizing the sum of squared errors. The solution for \mathbf{R} , however, is only guaranteed to

be a unitary matrix and therefore, can yield a rotation and reflection rather than only rotation. This issue is addressed in Sect. 3.2.

2.2 The UME–Universal Manifold Embedding

In this section, we briefly review the principles of the UME [27] for observations related by an affine transformation. Let \mathcal{O} be the space of observations. Let Φ be the group of affine transformations, and let S be a set of known objects. Every observation is the result of applying a geometric deformation $\phi \in \Phi$ to an object $s \in S$. The parameters of the affine transformation completely specify the action of the group of geometric transformations the object may undergo. We denote by $\phi s \subset \mathcal{O}$ the set of all possible observations on an object s . Thus, ϕs is the orbit of s under Φ .

The universal manifold embedding is a map $T: \mathcal{O} \rightarrow H$ from the space of observations into a low dimensional Euclidean space, H , such that the set $T(\phi s)$ is a linear subspace of H for any s . Thus, the UME reduces the dimension of any problem concerning the multiplicity of appearances of objects from the high dimensional space of observations \mathcal{O} to the low dimensional linear space H and allows for the usage of classical linear theory in solving the highly nonlinear problems of deformable object detection and registration.

Next, the mapping T is described: Consider the case where the finite support functions $h(\mathbf{x})$, $g(\mathbf{x})$ are observations on the same object related by an affine transformation, *i.e.*

$$h(\mathbf{x}) = g(\mathbf{Ax} + \mathbf{c}), \mathbf{A} \in GL(n), \mathbf{c}, \mathbf{x} \in \mathbb{R}^n \tag{3}$$

Let $\mathbf{y} = \mathbf{Ax} + \mathbf{c}$, $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y} + \mathbf{b}$ where $\mathbf{b} = -\mathbf{A}^{-1}\mathbf{c}$. Let $\mathbf{y}' = [1, \mathbf{y}^T]$ then $\mathbf{x} = \mathbf{Dy}'^T$ where $\mathbf{D} = [\mathbf{b}; \mathbf{A}^{-1}]$ is an $n \times (n + 1)$ matrix.

Also, let $P \in \mathbb{N}$ and let $w_l, l = 1, \dots, P$ be a set of bounded, Lebesgue measurable functions $w_l: \mathbb{R}^n \rightarrow \mathbb{R}$. Since by definition $h(\mathbf{x}) = 0, \mathbf{x} \notin \text{supp}(h)$, and similarly for g , by a change of variables we obtain the following identities:

$$\int_{\mathbb{R}^n} w_l \circ h(\mathbf{x})d\mathbf{x} = \left| \mathbf{A}^{-1} \right| \int_{\mathbb{R}^n} w_l \circ g(\mathbf{y})d\mathbf{y} \tag{4}$$

$$\int_{\mathbb{R}^n} \mathbf{x}^T w_l \circ h(\mathbf{x})d\mathbf{x} = \left| \mathbf{A}^{-1} \right| \int_{\mathbb{R}^n} \mathbf{y}' \mathbf{D}^T w_l \circ g(\mathbf{y})d\mathbf{y} \tag{5}$$

Let f be some observation on a deformable object. Its UME matrix is given by

$$\begin{aligned} \mathbf{T}(f) &= \begin{bmatrix} \int_{\mathbb{R}^n} w_1 \circ f(\mathbf{y})d\mathbf{y} & \int_{\mathbb{R}^n} y_1 w_1 \circ f(\mathbf{y})d\mathbf{y} & \dots & \int_{\mathbb{R}^n} y_n w_1 \circ f(\mathbf{y})d\mathbf{y} \\ \vdots & \vdots & \ddots & \vdots \\ \int_{\mathbb{R}^n} w_P \circ f(\mathbf{y})d\mathbf{y} & \int_{\mathbb{R}^n} y_1 w_P \circ f(\mathbf{y})d\mathbf{y} & \dots & \int_{\mathbb{R}^n} y_n w_P \circ f(\mathbf{y})d\mathbf{y} \end{bmatrix} \end{aligned} \tag{6}$$

Let $\mathbf{D}' = [\mathbf{e}_1; \mathbf{D}^T]$, where $\mathbf{e}_1 = [1 \ 0 \ \dots \ 0]^T$, be the matrix representation of an affine transformation in homogeneous coordinates. Rewriting (4), (5) for $l = 1, \dots, P$ in a matrix form by row stacking (4) and (5) for every l , and using the notation $\mathbf{G} = \mathbf{T}(g)$, $\mathbf{H} = \mathbf{T}(h)$, we have:

$$\mathbf{GD}' \left| \mathbf{A}^{-1} \right| = \mathbf{H}. \tag{7}$$

To find the matrix \mathbf{D}' (and thus recover the parameters of the affine transformation), we notice that (7) is in fact an over determined linear equation system. Hence, the least squares solution for \mathbf{D}' are given by

$$\mathbf{D}' = |\mathbf{A}| [\mathbf{G}^T \mathbf{G}]^{-1} \mathbf{G}^T \mathbf{H} \tag{8}$$

Remark 1 The set of matrices of the structure defined above for \mathbf{D}' is homomorphic to the affine group with matrix multiplication as the group operation.

3 The UME for Rigid Transformations

In the following, we derive the Rigid Transformation Universal Manifold Embedding (RT-UME) estimator of the rotation and translation relating two functions defined on \mathbb{R}^n . In Sect. 5, we consider the special case where $n = 3$ and the problem is the registration of 3D point clouds.

Assume that \mathcal{O} is the space of n dimensional functions from \mathbb{R}^n to \mathbb{R} . As an example, one may consider a finite support point cloud, where for each point in the point cloud we also measure the object gray-level value at this point. Assuming further that the objects undergo rigid geometric transformations of coordinates, Φ becomes the special Euclidean group in n dimensions, $SE(n)$. In [16], the Rigid Transformation UME for the noise-free case, derived in this section, is derived using a slightly different estimator.

Since $SE(n)$ is a subgroup of the affine group, the mapping $T: \mathcal{O} \rightarrow H$ in (6) is still valid. The coordinate transformation, however, is now different. As the geometric transformation is assumed to be rigid, the matrix \mathbf{A} becomes an orthonormal matrix with determinant of 1, *i.e.*, a rotation matrix. Let us denote this rotation matrix by \mathbf{R} . In this special case of a rigid deformation, \mathbf{D}' will be denoted by $\mathbf{D}'(\mathbf{R}, \mathbf{c})$ to distinguish the notation from the one used in the case of affine transformations.

Let $\mathbf{x}' = [1, \mathbf{x}^T]$. Using this homogeneous coordinates notation, we have $\mathbf{x}' = \mathbf{y}'\mathbf{D}'(\mathbf{R}, \mathbf{c})$, and hence, when

$$h(\mathbf{x}) = g(\mathbf{R}\mathbf{x} + \mathbf{c}) \tag{9}$$

(7) is rewritten in the following form:

$$\mathbf{G}\mathbf{D}'(\mathbf{R}, \mathbf{c}) = \mathbf{H} \tag{10}$$

where $\mathbf{D}'(\mathbf{R}, \mathbf{c})$ is given by:

$$\mathbf{D}'(\mathbf{R}, \mathbf{c}) = \begin{bmatrix} 1 & -\mathbf{c}^T \mathbf{R} \\ \mathbf{0} & \mathbf{R} \end{bmatrix} \tag{11}$$

and $\mathbf{b} = -\mathbf{R}^T \mathbf{c}$.

Thus, (10) establishes the basic relation of the Rigid Transformation UME (RT-UME). Clearly, a solution for $\mathbf{D}'(\mathbf{R}, \mathbf{c})$ cannot be found by applying the least squares solution given in (8), as there is no restriction on the solution to provide a rigid transformation. We next show that by enforcing the rigid transformation constraints in formalizing the least squares estimate of $\mathbf{D}'(\mathbf{R}, \mathbf{c})$ in (8), the solution is obtained along similar lines to the ones employed in [30]:

Define the error matrix $\mathbf{E} = \mathbf{G}\mathbf{D}'(\mathbf{R}, \mathbf{c}) - \mathbf{H}$. Let \mathbf{E}_i denote the i -th row of \mathbf{E} . Thus,

$$\mathbf{E}_i = \left[\begin{array}{c} \int_{\mathbb{R}^n} w_i \circ h(\mathbf{x}) d\mathbf{x} - \int_{\mathbb{R}^n} w_i \circ g(\mathbf{y}) d\mathbf{y} \\ \int_{\mathbb{R}^n} \mathbf{x} w_i \circ h(\mathbf{x}) d\mathbf{x} - \mathbf{R}^T \int_{\mathbb{R}^n} \mathbf{y} w_i \circ g(\mathbf{y}) d\mathbf{y} - \mathbf{b} \int_{\mathbb{R}^n} w_i \circ g(\mathbf{y}) d\mathbf{y} \end{array} \right]^T \tag{12}$$

and

$$\begin{aligned} \|\mathbf{E}_i\|^2 &= \left\| \int_{\mathbb{R}^n} w_i \circ h(\mathbf{x}) d\mathbf{x} - \int_{\mathbb{R}^n} w_i \circ g(\mathbf{y}) d\mathbf{y} \right\|^2 + \\ &+ \left\| \int_{\mathbb{R}^n} \mathbf{x} w_i \circ h(\mathbf{x}) d\mathbf{x} - \mathbf{R}^T \int_{\mathbb{R}^n} \mathbf{y} w_i \circ g(\mathbf{y}) d\mathbf{y} - \mathbf{b} \int_{\mathbb{R}^n} w_i \circ g(\mathbf{y}) d\mathbf{y} \right\|^2 \end{aligned} \tag{13}$$

The first term is independent of the deformation parameters; therefore, it is enough to minimize the sum over the second term. Define

$$\mathbf{g}_i = \int_{\mathbb{R}^n} \mathbf{y}^T w_i \circ g(\mathbf{y}) d\mathbf{y} \tag{14}$$

$$\mathbf{h}_i = \int_{\mathbb{R}^n} \mathbf{x}^T w_i \circ h(\mathbf{x}) d\mathbf{x} \tag{15}$$

$$m_i = \int_{\mathbb{R}^n} w_i \circ g(\mathbf{y}) d\mathbf{y} \tag{16}$$

Minimizing the sum of squared errors (13) for all $i = 1, \dots, P$ is equivalent to minimizing

$$\sum_{i=1}^P \left\| \mathbf{h}_i - \mathbf{g}_i \mathbf{R} - \mathbf{b}^T m_i \right\|^2 \tag{17}$$

This minimization highly resembles the minimization problem (2) solved in [30], with the exception that the translation term is multiplied by the mass coefficients m_i . The property of the translation *being identical for all i in (2)* is critical in the procedure of solving (2) in [30]. However, following [29], (17) can be rewritten as a weighted sum of squared errors,

$$\sum_{i=1}^P m_i^2 \left\| \frac{\mathbf{h}_i}{m_i} - \frac{\mathbf{g}_i}{m_i} \mathbf{R} - \mathbf{b}^T \right\|^2 \tag{18}$$

which is then solved for the desired \mathbf{R} and \mathbf{b} using the procedure described next.

3.1 Finding the Translation

Using the weighted centroid of \mathbf{h}_i and \mathbf{g}_i , we obtain their weighted-and-centered versions:

$$\mathbf{h}_i^c = \frac{\mathbf{h}_i}{m_i} - \frac{\sum_{j=1}^P m_j \mathbf{h}_j}{\sum_{j=1}^P m_j^2}, \quad \mathbf{g}_i^c = \frac{\mathbf{g}_i}{m_i} - \frac{\sum_{j=1}^P m_j \mathbf{g}_j}{\sum_{j=1}^P m_j^2} \tag{19}$$

In addition, define

$$\mathbf{b}^c = \mathbf{b}^T - \frac{\sum_{j=1}^P m_j \mathbf{h}_j}{\sum_{j=1}^P m_j^2} + \frac{\sum_{j=1}^P m_j \mathbf{g}_j}{\sum_{j=1}^P m_j^2} \mathbf{R} \tag{20}$$

Using these definitions, the sum in (18) can be written as

$$\sum_{i=1}^P m_i^2 \left\| \mathbf{h}_i^c - \mathbf{g}_i^c \mathbf{R} - \mathbf{b}^c \right\|^2 \tag{21}$$

Following similar lines to [30], it can be shown that $\mathbf{b}^c = 0$ minimizes (21). Therefore, substituting $\mathbf{b}^c = 0$ into (20) we have

$$\mathbf{b} = \left[\frac{\sum_{j=1}^P m_j \mathbf{h}_j}{\sum_{j=1}^P m_j^2} - \frac{\sum_{j=1}^P m_j \mathbf{g}_j}{\sum_{j=1}^P m_j^2} \mathbf{R} \right]^T \tag{22}$$

3.2 Estimating the Rotation

In order for \mathbf{R} to minimize (18), it is required to maximize

$$\sum_{i=1}^P m_i^2 \mathbf{h}_i^c \mathbf{R}^T (\mathbf{g}_i^c)^T = \text{tr} \{ \mathbf{R} \sum_{i=1}^P m_i^2 (\mathbf{h}_i^c)^T \mathbf{g}_i^c \} \quad (23)$$

Define

$$\mathbf{M} = \sum_{i=1}^P m_i^2 (\mathbf{h}_i^c)^T \mathbf{g}_i^c \quad (24)$$

Let $\mathbf{G}^c = [(\mathbf{g}_1^c)^T, \dots, (\mathbf{g}_p^c)^T]^T$, and \mathbf{H}^c is similarly defined. Rewriting the definition of \mathbf{M} in a matrix form, we have: $\mathbf{M} = (\mathbf{H}^c)^T \mathbf{W} \mathbf{G}^c$ where \mathbf{W} is a $P \times P$ diagonal weight matrix with the i -th term on the diagonal being m_i^2 . Note that \mathbf{M} is a square matrix. In contrast to the solution for \mathbf{R} in [30], where \mathbf{R} is only guaranteed to be unitary (i.e., a reflection may also occur), we adopt the solution method found in [4] and [40], where a rotation is guaranteed. Let $\mathbf{M} = \mathbf{U} \mathbf{S} \mathbf{V}^T$ the singular value decomposition of \mathbf{M} . From [4], the rotation matrix maximizing $\text{tr} \{ \mathbf{R} \mathbf{M} \}$ is given by

$$\mathbf{R} = \mathbf{V} \mathbf{C} \mathbf{U}^T, \quad \mathbf{C} = \text{diag} \{ 1, 1, \dots, 1, \det(\mathbf{U} \mathbf{V}^T) \} \quad (25)$$

Substituting the estimated \mathbf{R} into (22) completes the estimation of the rigid transformation.

4 Model Mismatch

In this section, it is shown that the presence of observation noise introduces bias to the constructed UME matrix. Then, given the noise statistics, an unbiased estimator is derived. In order to keep the generality of the derivation, it is presented for functions defined on \mathbb{R}^n , where the special case of \mathbb{R}^3 is easily concluded.

4.1 The Effect of Noise on the UME Matrix

Let $\tilde{h}(\mathbf{x}) = h(\mathbf{x}) + n(\mathbf{x})$ be a noisy observation, where $n(\mathbf{x})$ is the additive noise to the observation and $h(\mathbf{x})$ is a noise-free observation defined in (3). Following [27], it is concluded that the nonlinear left-compositions $\{w_i\}_{i=1}^P$ must be designed such that the result of their application is covariant with the geometric transformation, and hence, they cannot be functions of the coordinates. Hence, it is natural to choose the nonlinear $\{w_i\}_{i=1}^P$ functions as indicator functions of level-sets of $h(\mathbf{x})$. As a consequence, we perform the RT-UME noise analysis, presented next, for the case where the nonlinear left-compositions $\{w_i\}_{i=1}^P$ are the indicator function on

$h(\mathbf{x})$ such that

$$w_i(h(\mathbf{x})) = \mathbb{1}_{[l_i, h_i]}(h(\mathbf{x})) \quad (26)$$

where $l_i, h_i \in \mathbb{R}$ are the low and high boundary values for the i -th indicator function. Let $\tilde{\mathbf{H}}$ be the UME matrix constructed from a noisy observation $\tilde{h}(\mathbf{x})$. (Recall that \mathbf{H} and \mathbf{G} are the UME matrices constructed from a noise free observations $h(\mathbf{x})$ and $g(\mathbf{x})$). The i -th row of \mathbf{H} and similarly of $\tilde{\mathbf{H}}$ are therefore given by $\int_{\mathbb{R}^n} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(h(\mathbf{x})) d\mathbf{x}$ and $\int_{\mathbb{R}^n} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(\tilde{h}(\mathbf{x})) d\mathbf{x}$, respectively. Hence, in the presence of noise the sum of squared errors becomes

$$\sum_{i=1}^P \|\tilde{\mathbf{H}}_i - \mathbf{G}_i D'(\mathbf{R}, \mathbf{c})\|^2 \quad (27)$$

Proposition 1 Define the UME error matrix

$$\mathbf{e}_H = \tilde{\mathbf{H}} - \mathbf{H} \quad (28)$$

In addition, define the following sets:

$$X_{in}^{(i)} = \{ \mathbf{x} \in \text{supp}(h) \mid \mathbb{1}_{[l_i, h_i]}(h(\mathbf{x})) = 1, \mathbb{1}_{[l_i, h_i]}(\tilde{h}(\mathbf{x})) = 1 \} \quad (29)$$

$$X_{out}^{(i)} = \{ \mathbf{x} \in \text{supp}(h) \mid \mathbb{1}_{[l_i, h_i]}(h(\mathbf{x})) = 0, \mathbb{1}_{[l_i, h_i]}(\tilde{h}(\mathbf{x})) = 1 \} \quad (30)$$

$$X_{miss}^{(i)} = \{ \mathbf{x} \in \text{supp}(h) \mid \mathbb{1}_{[l_i, h_i]}(h(\mathbf{x})) = 1, \mathbb{1}_{[l_i, h_i]}(\tilde{h}(\mathbf{x})) = 0 \} \quad (31)$$

The i -th row of \mathbf{e}_H , denoted by $\mathbf{e}_{H,i}$ is given by

$$\mathbf{e}_{H,i} = \int_{X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} - \int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} \quad (32)$$

Proof See Appendix A

Therefore, the error in the UME matrix is caused by observation samples that, due to the noise, migrate across the boundaries of the indicator functions, such that they are included or excluded from the integration where they should not have. Figure 3 illustrates using the example of Fig. 2, an example of a correct integration area and the change in the integration area caused by noise.

In the following, it is shown that the presence of an additive observation noise results in bias of the least squares solution for the translation and rotation.

4.2 Mean UME Error Matrix

The analysis in Sect. 4.1 dealt with characterizing the error in a single realization of the observation. In this section,

we derive an expression for the mean UME matrix error $E(\mathbf{e}_H)$ given the unobservable noise free function $h(\mathbf{x})$ and the noise statistics. This expression is then used to “explain” and thus eliminate significant components from the noise related estimation errors. More specifically, it is shown that by embedding the statistical model in the constrained least squares solution, an unbiased estimator is achieved. In the following, it is assumed that the probability distribution of the additive noise is known. Therefore, one can find the probability of a certain $\mathbf{x} \in \text{supp}(h)$ to belong $X_{miss}^{(i)}$ or $X_{out}^{(i)}$. Let us denote the probability for a point \mathbf{x} to become an “outlier” point with respect to the function $h(\mathbf{x})$ and the noise model by $P_{out}^{(i)}(\mathbf{x}; h(\mathbf{x})) = P(\mathbf{x} \in X_{out}^{(i)})$ and similarly for a “missed” point $P_{miss}^{(i)}(\mathbf{x}; h(\mathbf{x})) = P(\mathbf{x} \in X_{miss}^{(i)})$.

Definition 1 The domain of $\mathbb{1}_{[l_i, h_i]}(h(\mathbf{x}))$ is characterized by the following sets

$$X_{above}^{(i)} = \{\mathbf{x} \in \text{supp}(h) | h_i < h(\mathbf{x})\} \tag{33}$$

$$X_{within}^{(i)} = \{\mathbf{x} \in \text{supp}(h) | l_i < h(\mathbf{x}) \leq h_i\} \tag{34}$$

$$X_{below}^{(i)} = \{\mathbf{x} \in \text{supp}(h) | h(\mathbf{x}) \leq l_i\} \tag{35}$$

Lemma 1 The mean UME error matrix $E(\mathbf{e}_H)$ is given by

$$E(\mathbf{e}_{H,i}) = \int_{X_{above}^{(i)} \cup X_{below}^{(i)}} \mathbf{x}' P_{out}^{(i)}(\mathbf{x}; h(\mathbf{x})) d\mathbf{x} - \int_{X_{within}^{(i)}} \mathbf{x}' P_{miss}^{(i)}(\mathbf{x}; h(\mathbf{x})) d\mathbf{x} \tag{36}$$

Proof See Appendix B.

Following Definition 1, let

$$Y_{above}^{(i)} = \{\mathbf{y} \in \text{supp}(g) | g(\mathbf{y}) > h_i\} \tag{37}$$

$$Y_{within}^{(i)} = \{\mathbf{y} \in \text{supp}(g) | l_i < g(\mathbf{y}) < h_i\} \tag{38}$$

$$Y_{below}^{(i)} = \{\mathbf{y} \in \text{supp}(g) | g(\mathbf{y}) < l_i\} \tag{39}$$

Since in general $E(\mathbf{e}_{H,i}) \neq 0$, we conclude that $\tilde{\mathbf{H}}$ is a biased estimate of \mathbf{H} . Assuming the noise distributions given $g(\mathbf{x})$ or $h(\mathbf{x})$, are identical, *i.e.*, the conditional probability distribution is independent of the observed function, we also have,

$$E(\mathbf{e}_{G,i}) := \int_{Y_{above}^{(i)} \cup Y_{below}^{(i)}} \mathbf{y}' P_{out}^{(i)}(\mathbf{y}; g(\mathbf{y})) d\mathbf{y} - \int_{Y_{within}^{(i)}} \mathbf{y}' P_{miss}^{(i)}(\mathbf{y}; g(\mathbf{y})) d\mathbf{y} \tag{40}$$

4.3 Compensating for the Bias

Next, it is shown that the mean estimated rotation and translation may be found using the mean UME observation matrix.

This in turn allows us to show that an unbiased estimator may be realized by evaluating a “reference” mean UME matrix, *i.e.* $\mathbf{G} + E(\mathbf{e}_G)$ and employing it in the least squares problem.

Proposition 2 Let $\tilde{\mathbf{h}}_i$, \mathbf{g}_i and m_i be defined as in (17). In addition, let

$$\{\hat{\mathbf{R}}, \hat{\mathbf{t}}\} = \underset{\{\mathbf{R}, \mathbf{t}\}}{\text{argmin}} \sum_{i=1}^P \|\tilde{\mathbf{h}}_i - \mathbf{g}_i \mathbf{R} - m_i \mathbf{t}\|^2 \tag{41}$$

$$\{\bar{\mathbf{R}}, \bar{\mathbf{t}}\} = \underset{\{\mathbf{R}, \mathbf{t}\}}{\text{argmin}} \sum_{i=1}^P \|E(\tilde{\mathbf{h}}_i) - \mathbf{g}_i \mathbf{R} - m_i \mathbf{t}\|^2 \tag{42}$$

then $E(\hat{\mathbf{R}}) = \bar{\mathbf{R}}$ and $E(\hat{\mathbf{t}}) = \bar{\mathbf{t}}$

Proof See Appendix C

Proposition 2 implies that the expectation of the estimated transformation parameters when they are estimated from the noisy UME matrix using (41) is equal to the estimates obtained from the mean UME matrix using (42). It is thus concluded that the estimation bias can be eliminated by subtracting $E(\mathbf{e}_H)$ from $\tilde{\mathbf{H}}$ in (41). However, $h(\mathbf{x})$ is unknown and therefore, $E(\mathbf{e}_H)$ is unknown as well. On the other hand, $E(\mathbf{e}_G)$ is known, and the relation between $E(\mathbf{e}_H)$ and $E(\mathbf{e}_G)$ is given next:

Proposition 3 Assuming that $h(\mathbf{x}) = g(\mathbf{R}\mathbf{x} + \mathbf{c})$, we have that

$$E(\mathbf{e}_{H,i}) = E(\mathbf{e}_{G,i}) \mathbf{D}'(\mathbf{R}, \mathbf{c}) \tag{43}$$

Proof Note that

$$Y_{above} = \mathbf{R} X_{above} + \mathbf{c} \tag{44}$$

$$Y_{within} = \mathbf{R} X_{within} + \mathbf{c} \tag{45}$$

$$Y_{below} = \mathbf{R} X_{below} + \mathbf{c} \tag{46}$$

By a change of variable $\mathbf{x}' = \mathbf{y}' \mathbf{D}'(\mathbf{R}, \mathbf{c})$ in (36)

$$\begin{aligned} E(\mathbf{e}_{H,i}) &= \int_{Y_{above}^{(i)} \cup Y_{below}^{(i)}} \mathbf{y}' \mathbf{D}'(\mathbf{R}, \mathbf{c}) P_{out}^{(i)}(\mathbf{y}; h(\mathbf{R}^T \mathbf{y} - \mathbf{R}^T \mathbf{c})) d\mathbf{y} \\ &\quad - \int_{Y_{within}^{(i)}} \mathbf{y}' \mathbf{D}'(\mathbf{R}, \mathbf{c}) P_{miss}^{(i)}(\mathbf{y}; h(\mathbf{R}^T \mathbf{y} - \mathbf{R}^T \mathbf{c})) d\mathbf{y} \\ &= \left(\int_{Y_{above}^{(i)} \cup Y_{below}^{(i)}} \mathbf{y}' P_{out}^{(i)}(\mathbf{y}; g(\mathbf{y})) d\mathbf{y} - \int_{Y_{within}^{(i)}} \mathbf{y}' P_{miss}^{(i)}(\mathbf{y}; g(\mathbf{y})) d\mathbf{y} \right) \mathbf{D}'(\mathbf{R}, \mathbf{c}) \\ &= E(\mathbf{e}_{G,i}) \mathbf{D}'(\mathbf{R}, \mathbf{c}) \end{aligned}$$

□

Corollary 1 Let $\tilde{\mathbf{H}}$ be the UME matrix of some noisy observation \tilde{h} and \mathbf{G} a known UME matrix of a noise-free observation g on the same object. Then, $E(\tilde{\mathbf{H}}) = (\mathbf{G} + E(\mathbf{e}_G))D'(\mathbf{R}, \mathbf{c})$

Proof Following (28), we have that $\mathbf{e}_H = \tilde{\mathbf{H}} - \mathbf{H}$. Hence, $E(\tilde{\mathbf{H}}) = \mathbf{H} + E(\mathbf{e}_H)$ \square

Corollary 1 implies that instead of subtracting $E(\mathbf{e}_H)$ from $\tilde{\mathbf{H}}$ in (27), the estimation bias can be eliminated by adding $E(\mathbf{e}_G)$ to \mathbf{G} . We can therefore replace (27) by a modified least squares problem that provides an unbiased RT-UME estimator:

$$\{\mathbf{R}_U, \mathbf{t}_U\} = \operatorname{argmin}_{\{\mathbf{R}', \mathbf{t}'\}} \sum_{i=1}^P \|\tilde{\mathbf{H}}_i - (\mathbf{G} + E(\mathbf{e}_{G,i}))D'(\mathbf{R}', \mathbf{t}')\|^2 \quad (47)$$

From corollary 1, we have

$$\mathbf{G} + E(\mathbf{e}_G) = E(\tilde{\mathbf{H}})D'(\mathbf{R}^{-1}, -\mathbf{R}^{-1}\mathbf{c}) \quad (48)$$

Substituting in (47), we get

$$\begin{aligned} \{\mathbf{R}_U, \mathbf{t}_U\} \\ = \operatorname{argmin}_{\{\mathbf{R}', \mathbf{t}'\}} \sum_{i=1}^P \|\tilde{\mathbf{H}}_i - E(\tilde{\mathbf{H}}_i)D'(\mathbf{R}^{-1}, -\mathbf{R}^{-1}\mathbf{c})D'(\mathbf{R}', \mathbf{t}')\|^2 \end{aligned} \quad (49)$$

Then, following Remark 1, by change of variables where

$$D'(\mathbf{R}'', \mathbf{t}'') = D'(\mathbf{R}^{-1}, -\mathbf{R}^{-1}\mathbf{c})D'(\mathbf{R}', \mathbf{t}') \quad (50)$$

(49) is rewritten as

$$\{\mathbf{R}'_U, \mathbf{t}'_U\} = \operatorname{argmin}_{\{\mathbf{R}'', \mathbf{t}''\}} \sum_{i=1}^P \|\tilde{\mathbf{H}}_i - E(\tilde{\mathbf{H}}_i)D'(\mathbf{R}'', \mathbf{t}'')\|^2 \quad (51)$$

thus providing an equivalent minimization problem. Applying proposition 2 to this case, we obtain $E(\mathbf{R}'_U) = \mathbf{I}_n$ and $E(\mathbf{t}'_U) = \mathbf{0}$. However, from (50) $\mathbf{R}'_U = \mathbf{R}^{-1}\mathbf{R}_U$ and $\mathbf{t}_U = \mathbf{t}'_U + \mathbf{R}_U\mathbf{R}^{-1}\mathbf{b}$. Therefore,

$$E(\mathbf{R}_U) = \mathbf{R} \quad (52)$$

$$E(\mathbf{t}_U) = \mathbf{b} \quad (53)$$

By evaluating a “corrected” mean UME matrix from the available noise free reference function and employing this result in the least squares problem, bias errors are compensated for and an unbiased estimator is achieved in (47).

5 Estimation of Rigid Transformations of Point Clouds

In this section, we present a novel approach to point cloud registration: Since point clouds are usually just sets of coordinates in \mathbb{R}^3 with no obvious functional relation imposed on them, the first step in the proposed framework is to define a function that assigns each point in the point cloud with a value (scalar or vector valued), invariant to the action of the transformation group. This “invariant coloring” of the point cloud enables, in the second stage the application of the RT-UME estimator for estimating the rigid transformation between two point clouds.

There are different methodologies for defining an SE(3)-invariant “coloring” function on a point cloud. In the examples discussed in the following, two specific choices of an SE(3)-invariant coloring function are evaluated: The first is the local curvature estimated using the volume curvature descriptor [21] or by the surface variation descriptor [26]. The second is the signed distance of the point from the plane passing through the point cloud center of mass, such that the plane normal is the axis of smallest variation of the point cloud. (The axis is found using principal component analysis.) These choices of SE(3)-invariant coloring functions are examples, and other alternatives can be formulated. In fact, any invariant point cloud descriptor can be used. See an illustration using the signed distance and the volume curvature descriptor in Fig. 2. Figure 3 depicts the integration error in the presence of additive noise, using the signed distance function. We note that this choice is highly suitable for processing aerial LIDAR scans as in those scans, the axis of smallest variation can be easily found and provides useful geometric information for constructing the UME matrices. This is demonstrated in the experiments presented in Sect. 6 where the proposed method achieves better performance than state-of-the-art methods when performing registration of aerial LIDAR scans.

The second challenge in adapting the RT-UME for processing point clouds is in evaluating the integrals defining the UME operator. Since the Lebesgue measure of the point cloud in \mathbb{R}^3 is zero, integration is performed using the counting measure instead, resulting with sums. Let $\mathcal{P} \subset \mathbb{R}^3$ be a point cloud, and $f(x) : \mathbb{R}^3 \mapsto \mathbb{R}$ be the SE(3)-invariant coloring function on the point cloud. In the special case where 3D point clouds are considered, (6) becomes

$$\begin{aligned} \mathbf{T}(\mathcal{P}) \\ = \begin{bmatrix} \sum_{\mathbf{x} \in \mathcal{P}} w_1 \circ f(\mathbf{x}) & \sum_{\mathbf{x} \in \mathcal{P}} x_1 w_1 \circ f(\mathbf{x}) & \dots & \sum_{\mathbf{x} \in \mathcal{P}} x_3 w_1 \circ f(\mathbf{x}) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{\mathbf{x} \in \mathcal{P}} w_P \circ f(\mathbf{x}) & \sum_{\mathbf{x} \in \mathcal{P}} x_1 w_P \circ f(\mathbf{x}) & \dots & \sum_{\mathbf{x} \in \mathcal{P}} x_3 w_P \circ f(\mathbf{x}) \end{bmatrix} \end{aligned} \quad (54)$$

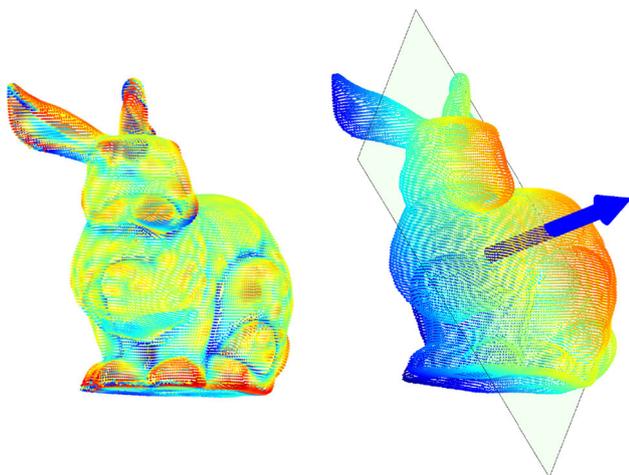


Fig. 2 SE(3)-invariant coloring functions for the construction of a UME matrix (54). Left: using the volume curvature descriptor [21]. Right: using the signed distance function. The function maps each point to the signed distance of the point from the plane passing through the point cloud center of mass, such that the plane normal is the axis of the smallest variation of the point cloud

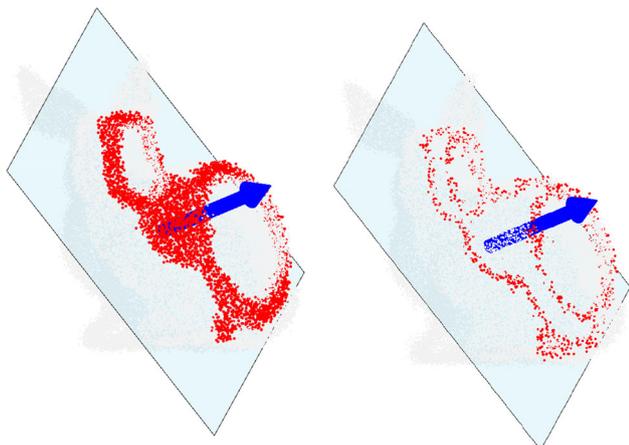


Fig. 3 An example of correct and erroneous integration due to additive noise (32). Left: Correct part of the integration area. Right: Erroneous part of the integration area

The complete registration process for 3D point clouds is depicted in Fig. 1 and described next:

Given two point clouds \mathcal{Q} and \mathcal{P} of the same object, related by a rigid transformation and possibly noise, the transformation parameters are estimated using the following procedure. First, an SE(3)-invariant coloring function is applied to the two point clouds. Next, corresponding RT-UME matrices, $\mathbf{H}_{\mathcal{Q}}$ and $\mathbf{H}_{\mathcal{P}}$, are constructed for each point cloud using (54). Since the transformation between \mathcal{Q} and \mathcal{P} is preserved by their corresponding UME matrices (10), the transformation relating the point clouds is estimated using (25) and (22) or by solving (47) when observation noise is present. Figure 4 provides several examples of registration between noisy observations and reference point clouds.

6 Experimental Results on Topographic Point Clouds

The experiments were conducted in several steps, presented here. At each of the following steps of the experiments, the performance of the presented method was compared with state-of-the-art baseline methods for varying noise and sampling rates. First, basic experiments validating the capability of the proposed method to register point clouds of topographic maps using the process described in Sect. 5 were conducted. In these experiments, the point clouds are of the same observed area (*i.e.*, full overlap between objects); however, the relative pose is unknown. In the following, this setup is referred to as the complete object registration. This setup provides the basis for the more realistic scenario detailed further on, while demonstrating the advantage of the proposed method over state of the art for this type of data.

Next, a more realistic set up is given as an illustrative example, simulating the problem of registering an observed SfM topographic model to a reference topographic model. In these experiments, only the general area from which the SfM was acquired is known. It is assumed to be completely contained in the reference model (see Fig. 8). This setup is referred to as observation to reference registration. The observation to reference setup is further verified with real data experiments in Sect. 6.4.4.

In the following sections, the baseline for comparison is detailed, followed by details of the conducted experiments in the two scenarios. While the generation of the data sets is similar in the two setups, details of the data sets are given in each section separately for the sake of clarity.

6.1 Baseline

The presented method is compared with a number of state-of-the-art point cloud registration algorithms. The parameters of the baseline methods were tuned by trial and error to optimize performance while limiting the running times to a maximum of few minutes for each point cloud pair. The first algorithm is a recent variant of the ICP called the globally optimal ICP (GOICP) [56], and the GOICP is expected to achieve the same accuracy as ICP when ICP is given a good initial position. GOICP was employed with no trimming since in the complete object registration scenario, full overlap is assumed between models, and similarly, in the observation to reference registration task, the moving point cloud is assumed to be fully contained in the reference. The convergence threshold was selected per noise level such that running times are no more than a few minutes for each pair of point clouds.

The second algorithm is the Super4PCS [38]. Super4PCS is a state-of-the-art algorithm which is independent of the initial pose and works well even when only little overlap is available, as reported by the authors. The Super4PCS algo-

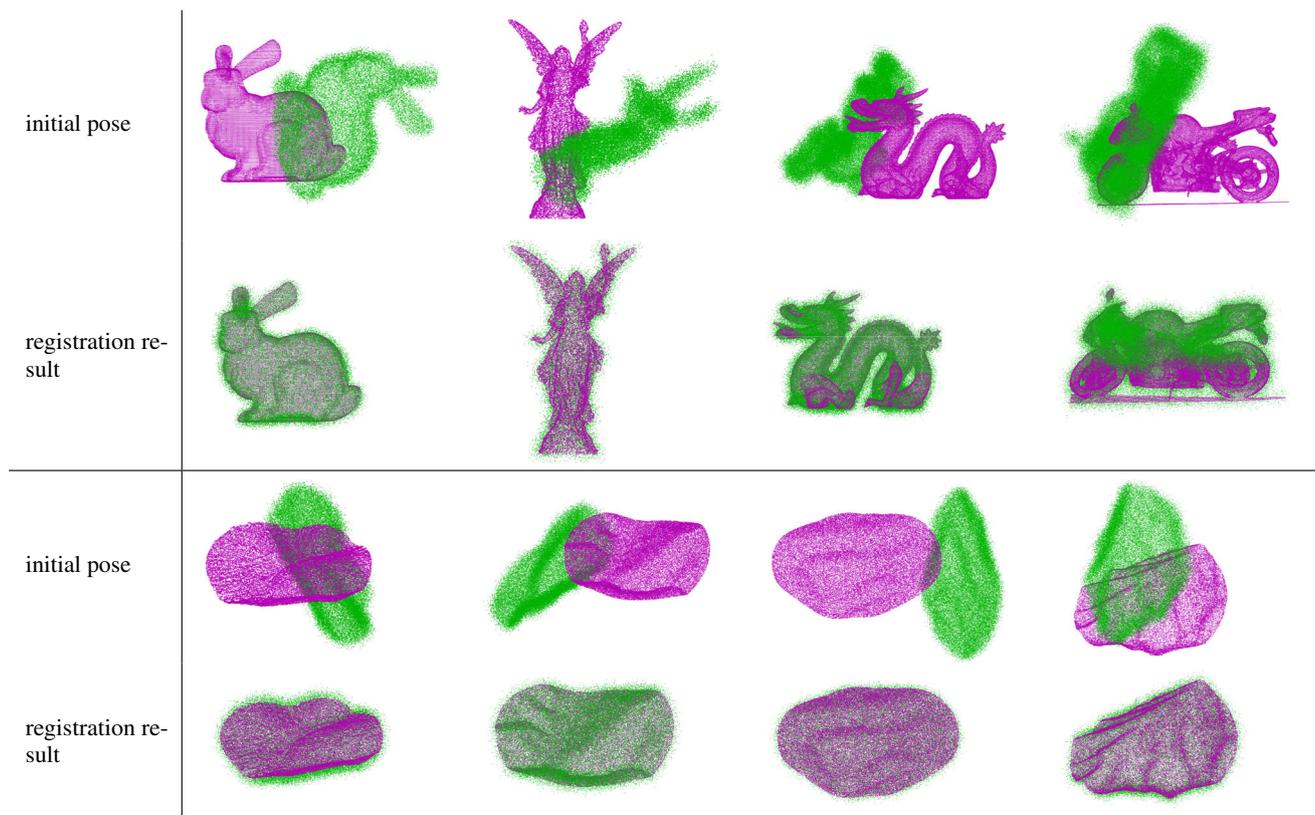


Fig. 4 Registration of noisy point clouds to a reference model using the unbiased RT-UME estimator (Sect. 4.3). Top, from left to right: Bunny, Lucy and Dragon models from "The Stanford 3D Scanning Repository", and a bike model from [11]. Bottom: examples of topographic point clouds used in Sect. 6

rithm creates correspondences based on congruent sets of co-planar four points sets and performs registration based on these correspondences. For the complete object registration, Super4PCS was run with 4 minutes time limit, down sampling was set to 2000 points and the overlap parameter was set to 0.5. For the observation to reference registration, time limit was set to 10 minutes, down sampling was set to 9000 points and the overlap parameter was set to 0.3. The overlap parameter in Super4PCS affects the size of the congruent sets, and we found it performs better with this setting rather than assuming full overlap as in GOICP.

The third algorithm is the JRMPC [14] where registration is achieved by approximating the surface using Gaussian mixture models. While several GMM-based algorithms exist (e.g., [39]), JRMPC does not assume the GMM centers are the points of the point clouds, significantly improving the computational requirements (when dealing with high-resolution LIDAR data the computational requirements of [39] become prohibitive). JRMPC was initialized with random points, related to the size of models. Maximum iterations were set to 100.

The fourth algorithm is a standard key point matching algorithm implemented in Open3D [63]. Key points are

found by down sampling the point clouds, and then, FPFH [46] descriptors are computed to create an initial set of matches in the down sampled point clouds, followed by an extensive RANSAC procedure.

The registration result of each baseline algorithm as well as of the RT-UME is then used as an initialization for ICP registration. ICP is by far the most commonly used algorithm for registration once an initial alignment is available. Since feature extraction and matching algorithms (e.g., [21,46,55]), and others (e.g., [37,38]), are complemented with ICP, this provides a fair and wide baseline. Moreover, the application of ICP to refine the initial registration by the compared methods provides insight as to how well each algorithm performs as an initializer for ICP and evaluates the overall performance achieved by refining the outcome of each of these algorithms using ICP. In the presented experiments, we use the MATLAB implementation of ICP, assuming full overlap and using point to plane metric.

6.2 Performance Evaluation

The registration performance of the tested methods is compared using two criteria: mean and standard deviation of the

absolute registration error (*i.e.*, rotation and translation estimation errors), and success rate. A registration is said to be successful if the rotation angle error is below 5° and if the centroid reprojection error is under 5 meters.

6.3 Complete Object Registration

In the basic experimental setup, the topographic models are of the same observed area (*i.e.*, there is full overlap between objects). Nevertheless, this setup proves to be challenging to state-of-the-art methods due to the relatively “flat” nature of the point clouds and their large size. As shown in the following examples, the proposed RT-UME method efficiently and accurately deals with these difficulties.

6.3.1 Data Sets

All the synthetic experiments were conducted on data generated from OpenTopography open source LIDAR database with the exception of the one data source that was acquired commercially. The LIDAR data sets [2,24,41,42,49,53] cover large areas with extremely high resolution. For this setup, more than hundred experimental data set pairs (referred to as the moving and fixed point clouds) were generated from the original LIDAR databases, each covering an area of approximately one square kilometer and re-sampled uniformly to 5 meter resolution. Data sets were chosen to represent a variety of terrain topologies such as high steep mountains, small hills and creeks. Figure 4 depicts examples of the data set used from each category together with generated data set pairs and registration outcome for this setup.

To evaluate the performance of the RT-UME, we present three types of experiments for this setup. First, the moving point cloud was misaligned by a random rigid transformation and corrupted with additive Gaussian noise with increasing standard deviation (Sect. 6.3.2). Then, the moving point cloud was misaligned by a random rigid transformation and randomly down sampled (Sect. 6.3.3). Finally, both down sampling and noise were applied (Sect. 6.3.4).

6.3.2 Performance in the Presence of Noise

In this part of the experiments, for each observation and reference pair, the observation was misaligned with a random rigid transformation and corrupted by an additive Gaussian noise. The tested standard deviation of the noise ranged between 0.5 and 10 meters. First, we compare the RT-UME with the unbiased RT-UME estimator derived in Sect. 4.3. In this comparison, Monte Carlo simulations were performed to estimate the error statistics. Using Fig. 5, we conclude that in the presence of a significant amount of noise, the average error and the success rate of the unbiased RT-UME estimator are considerably better than for the RT-UME.

Next, we compare the performance of the RT-UME to that of few popular baseline methods. In terms of average estimation error, it is concluded using Fig. 6 that for low noise levels, RT-UME, Super4PCS and FPFH+RANSAC registration perform similarly, while GOICP has a larger average error. However, as noise increases the accuracy decreases in all methods except the unbiased RT-UME where we experience only a small decrease in accuracy. Refining the results of all the methods by applying ICP provides performance improvement, except for GOICP where as expected no significant performance gain is achieved. The larger average error in GOICP can be explained by a small amount of badly failed registration results (see, Fig. 6c where GOICP has relatively high success rate).

In terms of success rate, it is concluded from the results depicted in Fig. 6c and f that the unbiased RT-UME performs better than the alternative methods. GOICP has slightly lower success rate. Moreover, using Fig. 6c and f we conclude that while without the ICP refinement stage, most methods fail to reach a successful registration, all methods (with the exception JRMPC) provide a good initial registration for ICP, as demonstrated by the jump in success rates after the ICP refinement. Throughout all experiments, the unbiased RT-UME outperforms the other methods while maintaining the high performance even when significant noise is present.

6.3.3 Performance with Random Down Sampling

In this part of the experiments, for each observation and reference pair, the observation was misaligned with a random rigid transformation and randomly down sampled. The tested down sampling rate ranges between 1 (no down sampling) to 0.05 of the original number of points. All but the RT-UME and JRMPC use a down sampling strategy as part of the algorithm. Therefore, as seen in Fig. 7 the accuracy of the baseline methods is similar for the varying sampling rates except at the lowest sampling rate of 0.1 and 0.05 where success rates and accuracy of all methods decrease. The RT-UME, which is an integral operator, is expected to be less accurate when sampling rates are low. Indeed its average error increases as the sampling rate decreases, however, its accuracy and success rates are better than those of the other methods up to the lowest sampling rate where most of the compared methods perform similarly. Comparing the accuracy with or without the ICP refinement stage, we conclude that with the exception of GOICP all methods benefit from applying ICP.

6.3.4 Performance with Down Sampling and Noise

In this part of the experiments, we present in detail a representative setup where the observations are randomly down sampled to 0.25 of the original number of points and the additive noise standard deviation is 10 meters. The

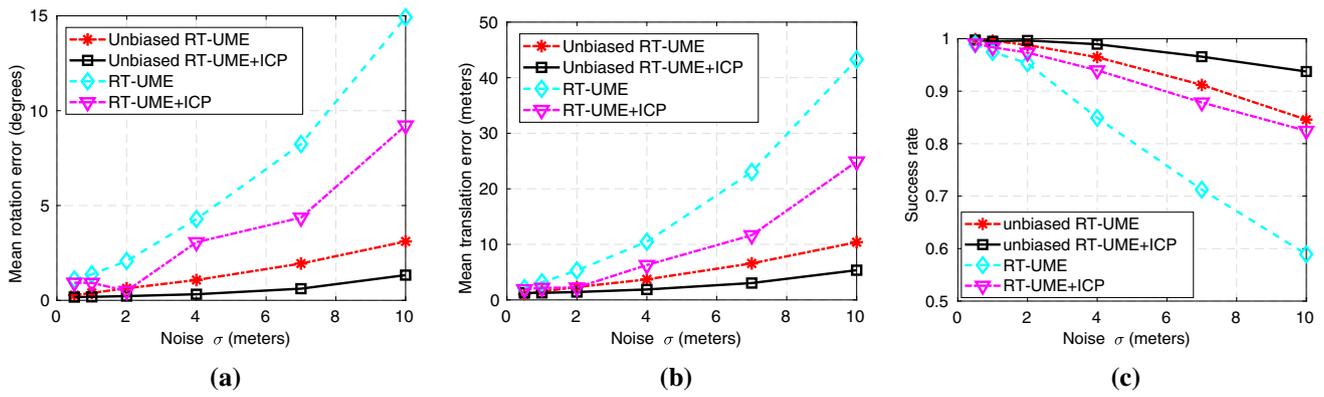


Fig. 5 RT-UME estimation mean of the absolute error and success rate as functions of observation noise standard deviation. (a) Rotation estimation error, (b) Translation estimation error and (c) success rate.

The unbiased RT-UME significantly outperforms the RT-UME at higher noise levels. Note that the unbiased RT-UME also provides a good initial alignment for ICP even at the highest noise levels

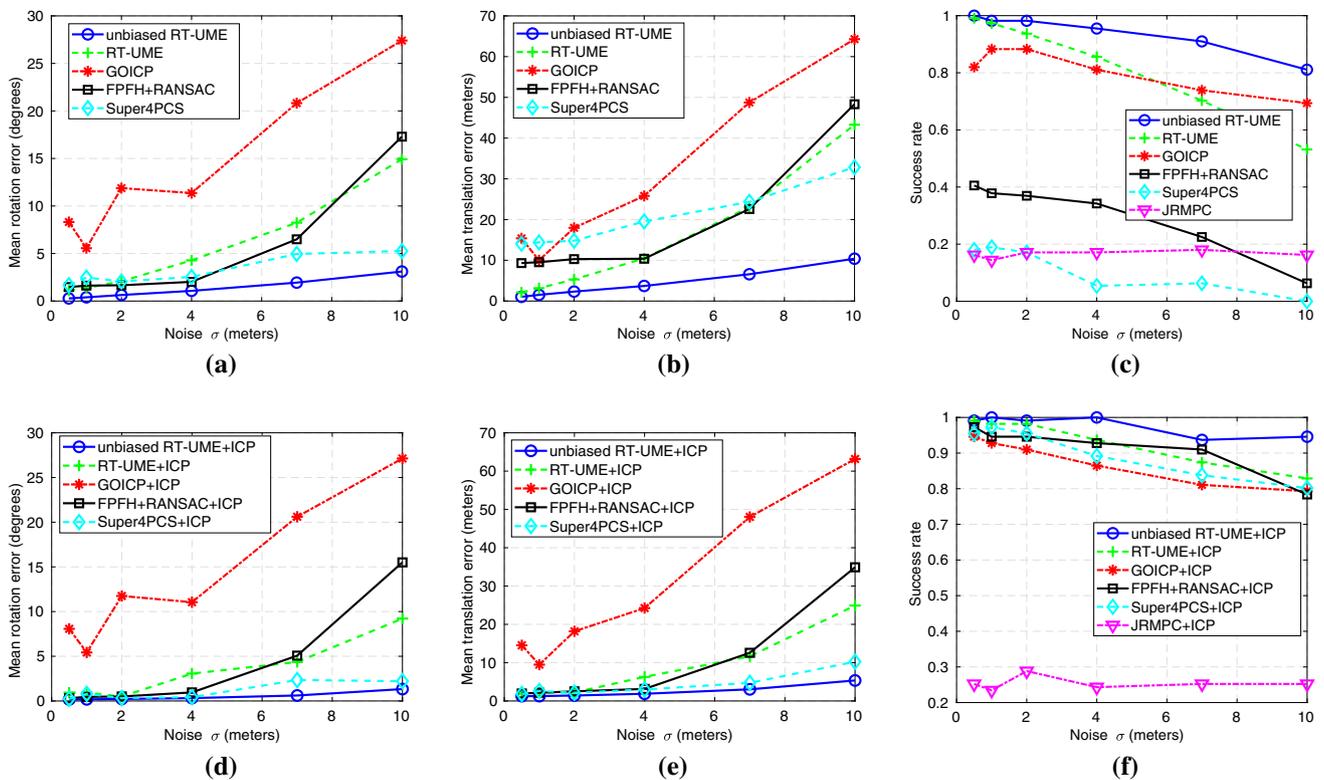


Fig. 6 RT-UME and baseline methods mean of the absolute error and success rates as functions of observation noise standard deviation with and without ICP. (a, d) Rotation estimation error, (b, e) Translation estimation error and (c, f) success rate. The unbiased RT-UME performs better than the alternatives at higher noise levels. Compared with the

baseline methods, ICP has little contribution to the unbiased RT-UME. The large increase in success rate for the baseline methods indicates that while being less accurate than the RT-UME they are capable of properly initializing the ICP algorithm. JRMPC was excluded from the plots due to the high average error

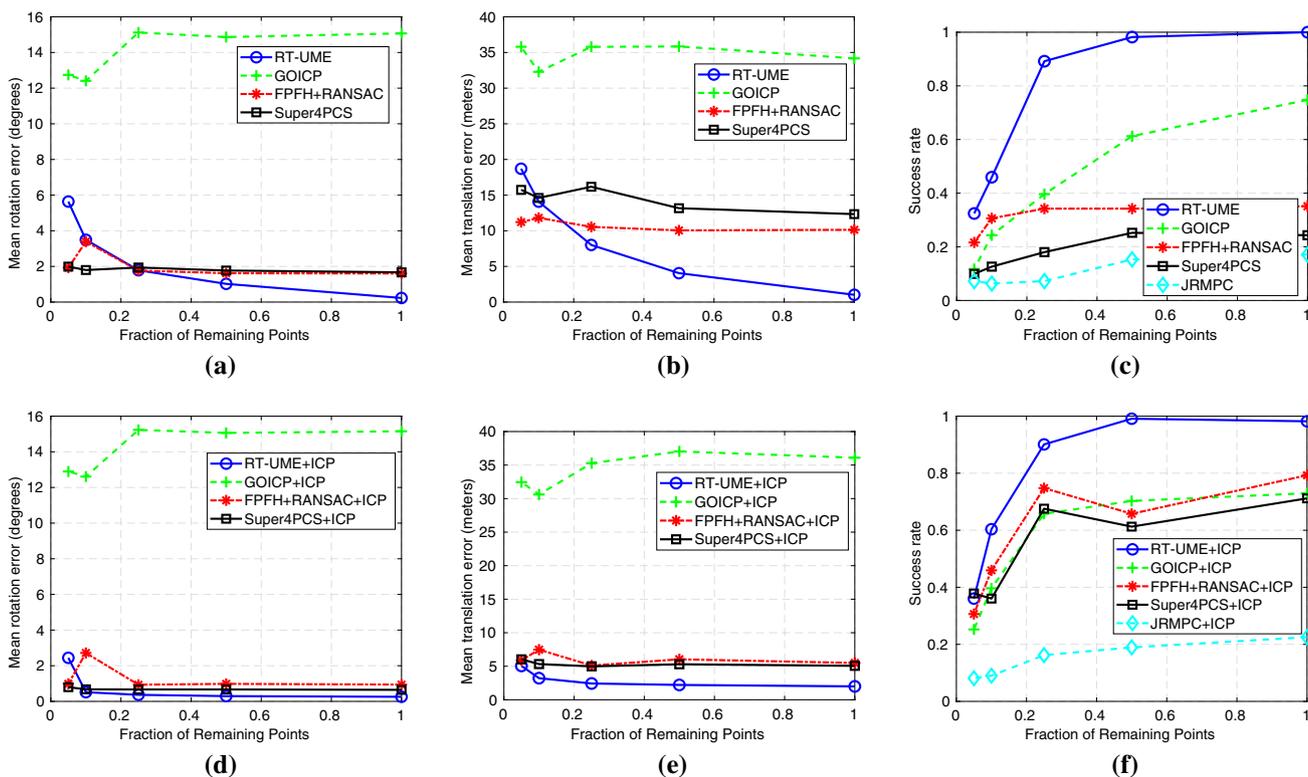


Fig. 7 RT-UME and baseline methods mean of the absolute error and success rate as functions of observation down sampling rates with and without ICP. (a, d) Rotation estimation error, (b, e) Translation estimation error and (c, f) success rate. The unbiased RT-UME outperforms the compared methods up to the lowest sampling rates where all meth-

ods perform similarly. The large increase in success rate for the baseline methods when followed by an ICP refinement stage indicates that while being less accurate than the RT-UME, they are useful to properly initialize the ICP algorithm. JRMPC was excluded from the plots due to the high average error

results are presented in Table 1. In this setup, the RT-UME performed best. In fact, in terms of success rate and translation estimation accuracy the performance of the unbiased RT-UME on its own is better than after performing ICP. GOICP performance was the second best. Super4PCS and FPFH+RANSAC gained most by performing ICP, which is consistent with the previous experiments where the success rates before ICP were very low, but increased significantly after employing ICP.

The high success rate of the RT-UME together with the low computational complexity provides the basis for the next step, where the observed area of the model is searched within the larger reference.

6.4 Observation to Reference Registration

In this section, we demonstrate the applicability of the registration method derived using the RT-UME to the problem of registration of an observed SfM topographic model to a reference topographic model. It is assumed that the observation is completely contained in the reference model (see Fig. 8). The observed model has an independent coordinate system,

and therefore, its location and pose relative to the reference model are unknown. In addition, the observation is assumed to be noisy, with different sampling rate than the reference model.

Unlike the first scenario discussed in Sect. 6.3, where both models were obtained from the same observed area, in order to apply the method described in Sect. 5 in the current setup, a simple search of the observed model within the reference model is employed. The search method is as follows. Let \mathcal{Q} and \mathcal{P} denote the observed and reference models, respectively. First crop a portion of \mathcal{Q} within a ball of radius r , denoted by \mathcal{Q}_r . This step allows for the resulting shape boundaries to be invariant to 3D rotations. At each point $\mathbf{p} \in \mathcal{P}$ of the reference model, crop a portion of the surface within a ball of the same radius r , denoted by \mathcal{P}_r^p . Next, registration between \mathcal{Q}_r and \mathcal{P}_r^p is performed as described in (5). Since the orientation of the model is unknown, the UME matrices registration MSE (17), which is invariant of the initial pose, is used as a fitting metric to determine the best fit between \mathcal{P}_r^p and \mathcal{Q}_r . The construction of the UME matrix is of linear computational complexity; thus, this search approach is fast. With very large data sets, a

Table 1 Complete object registration results of the compared methods. Observation was down sampled to 0.25 of the original number of points and corrupted by Gaussian noise with 10 meters standard deviation

Method	Rotation Error Mean (deg)	Rotation Error Standard deviation (deg)	Translation Error Mean (m)	Translation Error Standard deviation(m)	Success rate (%)
Unbiased RT-UME	2.06	2.25	2.05	1.23	89.19
Unbiased RT-UME+ICP	0.74	1.15	2.73	1.60	87.39
RT-UME	16.53	34.39	3.61	2.83	53.15
RT-UME+ICP	11.15	35.87	3.27	3.76	73.87
GOICP	2.75	17.15	4.02	2.87	68.46
GOICP+ICP	2.71	17.16	3.93	3.23	75.67
Super4PCS	8.44	25.60	21.10	15.34	1.80
Super4PCS+ICP	5.87	26.11	5.60	3.71	53.15
FPFH +RNASAC	15.79	43.35	16.22	13.92	9.00
FPFH +RANSAC+ICP	14.43	43.86	6.86	8.13	50.45
JRMPC	90.89	65.21	18.14	17.00	16.21
JRMPC+ICP	89.84	69.91	21.93	21.31	20.72

Bold values indicate the best achieved performance in each category.

faster search is achieved by employing an hierarchical search, first searching over a rough grid, and then refining the search near the best matching location (see Fig. 8c). The registration result of all compared methods was used as a starting point to ICP, similarly to Sect. 6.3.4. This gives insight both on the accuracy of each method on its own, and to how well it performs as initialization to ICP for an accurate final registration.

6.4.1 Synthetic Data sets

The generation of synthetic data sets for this setup is identical to the generation of synthetic data sets for complete object registration (See Sect. 6.3.4), except that in the present example the reference is approximately four times the area of the observation. The simulated observed SfM model covers a portion of the area covered by each reference model, it is randomly oriented and centered to the origin such that it is detached from the original coordinate system. Both generated reference maps and observed models were down sampled to

lower resolutions from the original LIDAR data to simulate different sampling points and densities. Figure 8 depicts an example of generated data sets: The observations were randomly down sampled to 0.25 of the original number of points and corrupted by additive Gaussian noise with 5 meters standard deviation.

6.4.2 Synthetic Data Results

The comparison criteria of the complete object registration experiments were used here as well (see Sect. 6.3.4). Table 2 summarizes the results. In terms of average estimation error, the unbiased RT-UME provided the best results with the lowest average error. Before employing ICP, GOICP provides the highest success rate. However, as expected, employing an ICP refinement stage does not provide any additional gain. On the other hand, the unbiased RT-UME followed by employing the ICP refinement, reached more than 88% success rate. This indicates that while the RT-UME success rate is not as high before employing ICP, it provides an adequate

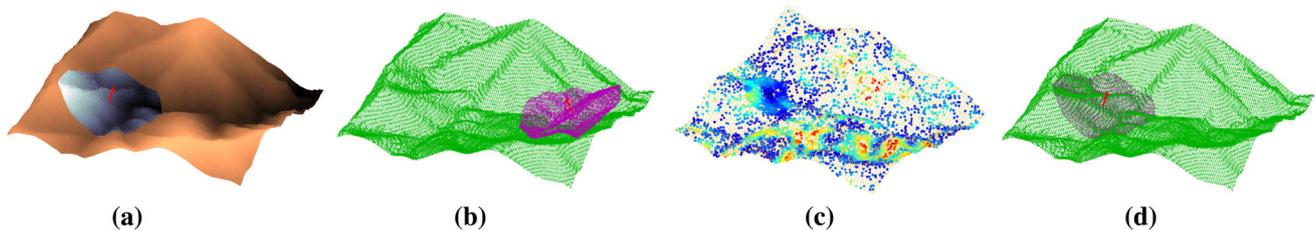


Fig. 8 (a) Reference point cloud with the observation at the correct position (b) Reference and observation point cloud initial pose (c) Registration error map over the search points, darker color indicates

lower registration error (d) Final alignment after applying the RT-UME. Alignment error: 2.17° , reprojection error: 2.8 meters

Table 2 Observation to reference registration results of the compared methods

Method	Rotation Error Mean (deg)	Rotation Error Standard deviation (deg)	Translation Error Mean (m)	Translation Error Standard deviation(m)	Success rate
unbiased RT-UME	7.12	14.32	17.59	30.55	20%
unbiased RT-UME+ICP	2.40	14.00	6.99	28.31	88.57%
GOICP	52.32	75.32	74.07	105.03	39.05%
GOICP+ICP	52.21	75.38	72.68	105.24	47.62%
Super4PCS	66.68	76.82	163.29	154.31	2.85%
Super4PCS+ICP	65.86	78.79	140.05	144.90	23.81%
FPFH	45.26	59.25	102.23	99.91	0.00%
FPFH+ICP	41.67	62.85	84.85	104.28	32.38%
JRMPC	89.21	56.74	142.29	93.87	0.00%
JRMPC+ICP	88.12	61.41	141.45	101.79	5.71%

Bold values indicate the best achieved performance in each category.

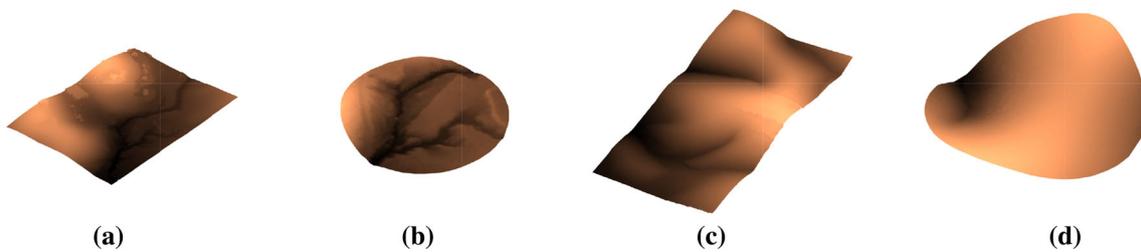


Fig. 9 (a) First real data reference point cloud (b) First real data observation point cloud (c) Second real data reference point cloud (d) second real data observation point cloud

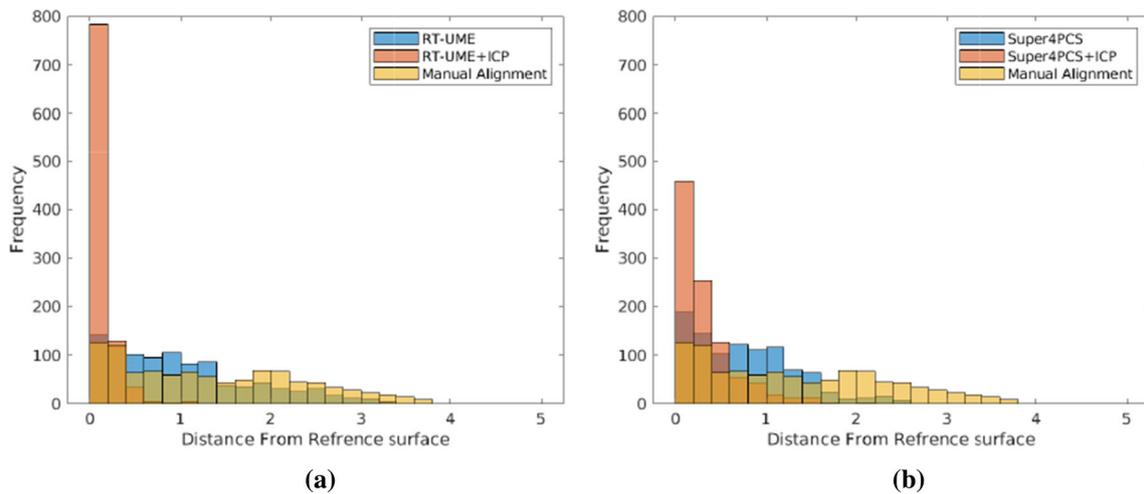


Fig. 10 Point to surface error histogram of the first model (Fig. 9a). (a) RT-UME and RT-UME+ICP vs Manual alignment (b) Super4PCS and Super4PCS+ICP vs Manual alignment. While providing similar fit before ICP, the RT-UME clearly provides a better initialization for ICP

D Real Data Registration Results

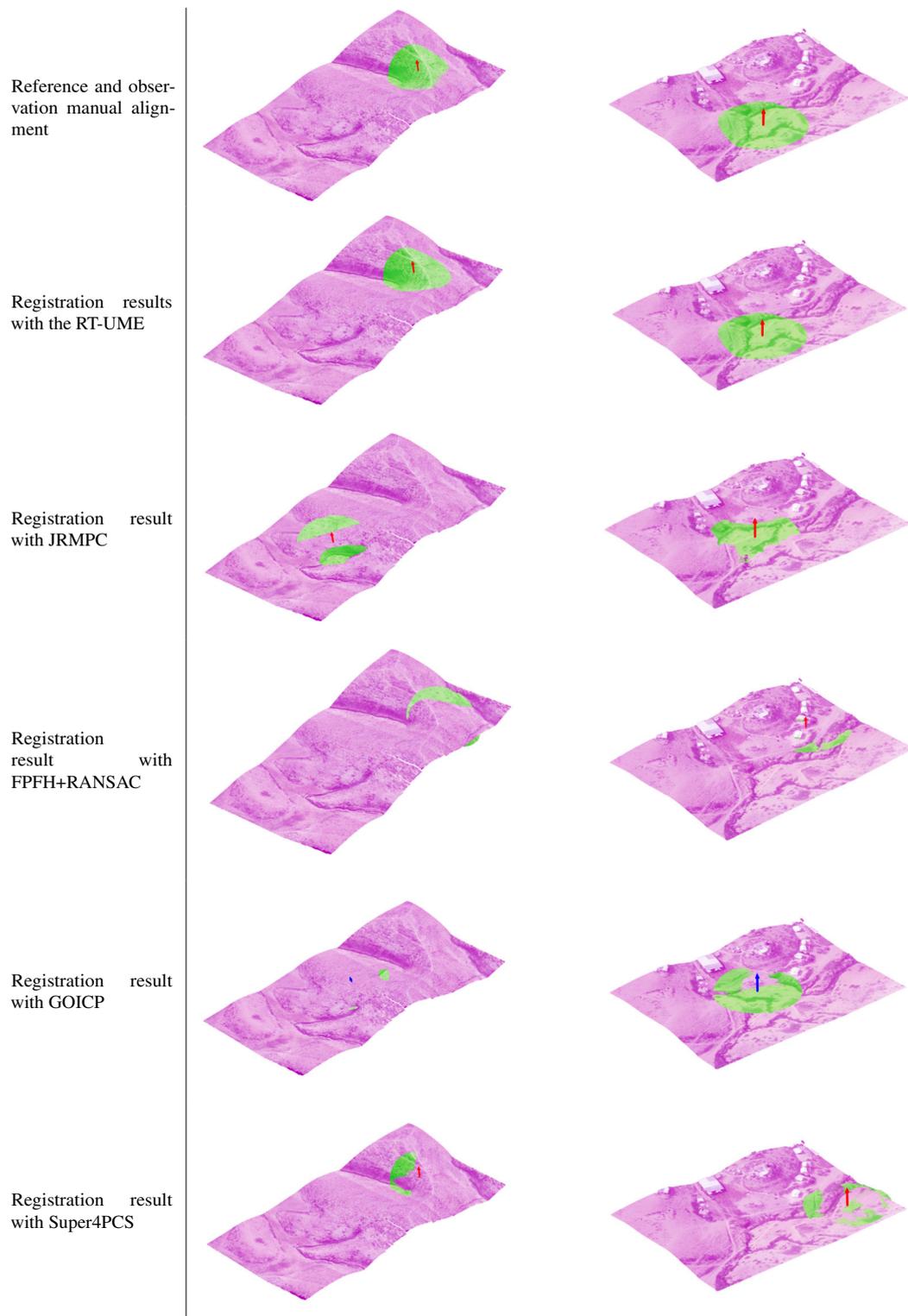


Fig. 11 Registration results of the compared methods with real data sets. The larger, purple model is the reference map, while the green point cloud is the SfM model. On the first row, the ground truth obtained by

manual alignment is presented, while in the rows below the registration results with various methods are depicted

initial registration for ICP. The large increase in success rate after employing ICP is demonstrated by FPFH+RANSAC and Super4PCS, as well. It is interesting to note that while the average estimation error after employing ICP decreases for most methods, the variance of estimation errors increase in some cases. This is due to the fact that when the initial registration fails, employing ICP can increase the estimation error and contribute to larger estimation variance.

6.4.3 Real Data Sets

For real data experiments, a high-resolution DSM commercial model (see, Fig. 9a and c) was used as a reference model. Two SfM models generated with Multi-View Environment [19] were used as observations. Each of the SfM models was generated from an aerial sequence of images with different camera setups resulting with the models seen in Fig. 9b and d. Ground truth estimation was extracted manually. The observed SfM model was also manually scaled before attempting registration with any of the applied methods, in order to separate the scale estimation problem from that of estimating the rigid transformation.

6.4.4 Real Data Results

For the real data experiments, an additional criteria for performance evaluation is employed: The point to surface distance between the registered observation and the reference was averaged over 1000 randomly selected points. The RT-UME method was successful in the registration of both models, while none of the baseline methods succeeded with the exception of Super4PCS being successful in initializing ICP. Figure 11 in Appendix D summarizes the registration results of the real data relative to a manual alignment. Figure 10 provides a comparison between the point to surface registration error histograms of the RT-UME-based registration, Super4PCS and manual alignment with and without ICP. While without ICP refinement, the distance between the registered observation to the reference was similar for RT-UME and Super4PCS, it is shown in Fig. 10 that the RT-UME provides a better initialization for ICP. These results further verify the ability of the RT-UME to perform fast, robust and accurate registration of topographic point clouds, where an observation is contained within a larger reference model.

7 Conclusions

We have presented the RT-UME, a closed form solution to the problem of registration of fully overlapping 3D point clouds undergoing unknown rigid transformations, as well as for detection and registration of sub-parts undergoing unknown rigid transformations. The solution is obtained by adapting

the general framework of the universal manifold embedding to the case where the transformations the object may undergo are rigid. For the case where the observations are subject to an additive noise with known parameters, an unbiased estimation scheme is derived. The experimental results, performed both on synthetic and real data, indicate that the proposed RT-UME provides higher registration accuracy than state-of-the-art methods, while both eliminating the need for an initial registration and reducing the computational complexity.

Acknowledgements This work is based on data services provided by the OpenTopography Facility with support from the National Science Foundation under NSF Award Numbers 1557484, 1557319, and 1557330.

A Proof for proposition 1

Proof First, notice that

$$\int_{\mathbb{R}^n} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(h(\mathbf{x})) d\mathbf{x} = \int_{X_{in}^{(i)} \cup X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} = \int_{X_{in}^{(i)}} \mathbf{x}' d\mathbf{x} + \int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} \tag{55}$$

$$\int_{\mathbb{R}^n} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(\tilde{h}(\mathbf{x})) d\mathbf{x} = \int_{X_{in}^{(i)} \cup X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} = \int_{X_{in}^{(i)}} \mathbf{x}' d\mathbf{x} + \int_{X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} \tag{56}$$

Adding and subtracting $\int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x}$ to (56), we obtain

$$\begin{aligned} & \int_{\mathbb{R}^n} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(\tilde{h}(\mathbf{x})) d\mathbf{x} \\ &= \int_{X_{in}^{(i)}} \mathbf{x}' d\mathbf{x} + \int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} - \int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} + \int_{X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} \end{aligned} \tag{57}$$

substituting (55) in (57)

$$\begin{aligned} & \int_{\mathbb{R}^n} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(\tilde{h}(\mathbf{x})) d\mathbf{x} = \int_{\mathbb{R}^n} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(h(\mathbf{x})) d\mathbf{x} \\ & - \int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} + \int_{X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} \end{aligned} \tag{58}$$

Hence,

$$\int_{\mathbb{R}^n} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(\tilde{h}(\mathbf{x})) d\mathbf{x} - \int_{\mathbb{R}^2} \mathbf{x}' \mathbb{1}_{[l_i, h_i]}(h(\mathbf{x})) d\mathbf{x} =$$

$$\int_{X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} - \int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} \tag{59}$$

□

B Mean UME Error Matrix Detailed Derivation

Following definition 1, we notice that

$$\int_{X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} = \int_{X_{above}^{(i)} \cup X_{below}^{(i)}} \mathbf{x}' \mathbb{1}_{[l_i-h(\mathbf{x}), h_i-h(\mathbf{x})]}(n(\mathbf{x})) d\mathbf{x} \tag{60}$$

and similarly

$$\int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} = \int_{X_{within}^{(i)}} \mathbf{x}' \mathbb{1}_{[h_i-h(\mathbf{x}), \infty]}(n(\mathbf{x})) d\mathbf{x} + \mathbf{x}' \mathbb{1}_{[-\infty, l_i-h(\mathbf{x})]}(n(\mathbf{x})) d\mathbf{x} \tag{61}$$

Therefore,

$$\begin{aligned} E \left(\int_{X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} \right) &= \int_{X_{above}^{(i)} \cup X_{below}^{(i)}} \mathbf{x}' E(\mathbb{1}_{[l_i-h(\mathbf{x}), h_i-h(\mathbf{x})]}(n(\mathbf{x}))) d\mathbf{x} \\ &= \int_{X_{above}^{(i)} \cup X_{below}^{(i)}} \mathbf{x}' P(l_i - h(\mathbf{x}) < n(\mathbf{x}) < h_i - h(\mathbf{x})) d\mathbf{x} \\ &= \int_{X_{above}^{(i)} \cup X_{below}^{(i)}} \mathbf{x}' P_{out}^{(i)}(\mathbf{x}; h(\mathbf{x})) d\mathbf{x} \end{aligned} \tag{62}$$

similarly it is found that

$$E \left(\int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} \right) = \int_{X_{within}^{(i)}} \mathbf{x}' P_{miss}^{(i)}(\mathbf{x}; h(\mathbf{x})) d\mathbf{x} \tag{63}$$

Substituting (63) and (62) into (32) and taking the expectation

$$\begin{aligned} E(\mathbf{e}_{H,i}) &= E \left(\int_{X_{out}^{(i)}} \mathbf{x}' d\mathbf{x} - \int_{X_{miss}^{(i)}} \mathbf{x}' d\mathbf{x} \right) \\ &= \int_{X_{above}^{(i)} \cup X_{below}^{(i)}} \mathbf{x}' P_{out}^{(i)}(\mathbf{x}; h(\mathbf{x})) d\mathbf{x} - \int_{X_{within}^{(i)}} \mathbf{x}' P_{miss}^{(i)}(\mathbf{x}; h(\mathbf{x})) d\mathbf{x} \end{aligned} \tag{64}$$

C Proof of Proposition 2

Proof The proof is a two step procedure. In the first step, the rotation between two sets of points is found by minimizing the sum of squared errors independently of the translation, showing that $E(\hat{\mathbf{R}}) = \bar{\mathbf{R}}$. Next, using the result for the rotation, it is shown that $E(\hat{\mathbf{t}}) = \bar{\mathbf{t}}$. Define $\tilde{\mathbf{h}}_i^c$ and \mathbf{g}_i^c , as in (19). From (21) and (22), the solutions for $\hat{\mathbf{R}}$ and $\bar{\mathbf{R}}$ in (41) and (42) can be written as

$$\hat{\mathbf{R}} = \operatorname{argmin}_{\mathbf{R}} \sum_{i=1}^P m_i^2 \|\tilde{\mathbf{h}}_i^c - \mathbf{g}_i^c \mathbf{R}\|^2 \tag{65}$$

$$\bar{\mathbf{R}} = \operatorname{argmin}_{\mathbf{R}} \sum_{i=1}^P m_i^2 \|E(\tilde{\mathbf{h}}_i^c) - \mathbf{g}_i^c \mathbf{R}\|^2 \tag{66}$$

Thus, the estimated rotation is determined by the centered UME matrices, and hence, it can be obtained independently of the estimated translation. In addition, from (22) we have that the translation is selected such that the estimation error of (41) and (42) is determined completely by (65) and (66). Define the estimation errors

$$\mathbf{e}_{res,i} = \tilde{\mathbf{h}}_i^c - \mathbf{g}_i^c \hat{\mathbf{R}} \tag{67}$$

$$\bar{\mathbf{e}}_{res,i} = E(\tilde{\mathbf{h}}_i^c) - \mathbf{g}_i^c \bar{\mathbf{R}} \tag{68}$$

Taking the expectation of both sides of (67), and substituting (67) into (68), we have

$$\mathbf{g}_i^c \bar{\mathbf{R}} + \bar{\mathbf{e}}_{res,i} = \mathbf{g}_i^c E(\hat{\mathbf{R}}) + E(\mathbf{e}_{res,i}) \tag{69}$$

Taking the m_i^2 weighted average of (69), we have

$$\begin{aligned} \frac{\sum_{i=1}^P m_i^2 \mathbf{g}_i^c E(\hat{\mathbf{R}}) + \sum_{i=1}^P m_i^2 E(\mathbf{e}_{res,i})}{\sum_{i=1}^P m_i^2} &= \frac{\sum_{i=1}^P m_i^2 \mathbf{g}_i^c \bar{\mathbf{R}} + \sum_{i=1}^P m_i^2 \bar{\mathbf{e}}_{res,i}}{\sum_{i=1}^P m_i^2} \end{aligned} \tag{70}$$

Since the solution is obtained using the weighted-and-centered versions $\tilde{\mathbf{h}}_i^c, \mathbf{g}_i^c$, it can be easily shown using the definition of the centering process in (19) that $\frac{\sum_{i=1}^P m_i^2 \mathbf{e}_{res,i}}{\sum_{i=1}^P m_i^2} = 0$ and $\frac{\sum_{i=1}^P m_i^2 \bar{\mathbf{e}}_{res,i}}{\sum_{i=1}^P m_i^2} = 0$. Thus, (70) becomes

$$\frac{\sum_{i=1}^P m_i^2 \mathbf{g}_i^c}{\sum_{i=1}^P m_i^2} E(\hat{\mathbf{R}}) = \frac{\sum_{i=1}^P m_i^2 \mathbf{g}_i^c}{\sum_{i=1}^P m_i^2} \bar{\mathbf{R}} \tag{71}$$

Assuming \mathbf{g}_i^c are not all zeros

$$E(\hat{\mathbf{R}}) = \bar{\mathbf{R}} \tag{72}$$

As for the translation, let us evaluate the mean translation from (41) by substituting (22):

$$\begin{aligned}
 E(\hat{\mathbf{t}}) &= E \left[\frac{\sum_{j=1}^P m_j \tilde{\mathbf{h}}_j}{\sum_{j=1}^P m_j^2} - \frac{\sum_{j=1}^P m_j \mathbf{g}_j}{\sum_{j=1}^P m_j^2} \hat{\mathbf{R}} \right]^T = \\
 &= \left[\frac{\sum_{j=1}^P m_j E(\tilde{\mathbf{h}}_j)}{\sum_{j=1}^P m_j^2} - \frac{\sum_{j=1}^P m_j \mathbf{g}_j}{\sum_{j=1}^P m_j^2} E(\hat{\mathbf{R}}) \right]^T = \\
 &= \left[\frac{\sum_{j=1}^P m_j E(\tilde{\mathbf{h}}_j)}{\sum_{j=1}^P m_j^2} - \frac{\sum_{j=1}^P m_j \mathbf{g}_j}{\sum_{j=1}^P m_j^2} \bar{\mathbf{R}} \right]^T = \bar{\mathbf{t}} \quad (73)
 \end{aligned}$$

where the last equality is due to the definition of (42). \square

D Real Data Registration Results

References

- Aiger, D., Mitra, N.J., Cohen-Or, D.: 4-points congruent sets for robust surface registration. *ACM Trans. Graph.* **27**(3), 1–10 (2008)
- Anderson, R.: Niwot ridge long-term ecological research site, colorado. National Center for Airborne Laser Mapping (NCALM), distributed by OpenTopography (2013)
- Aoki, Y., Goforth, H., Srivatsan, R.A., Lucey, S.: Pointnetlk: Robust & efficient point cloud registration using pointnet. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
- Arun, K., Huang, T., Blostein, S.: Least-squares fitting of two 3-d point sets. *IEEE Trans. Pattern Anal. Mach. Intell.* **9**, 698–700 (1987). <https://doi.org/10.1109/TPAMI.1987.4767965>
- Bai, X., Luo, Z., Zhou, L., Fu, H., Quan, L., Tai, C.L.: D3feat: Joint learning of dense detection and description of 3d local features. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6358–6366 (2020). <https://doi.org/10.1109/CVPR42600.2020.00639>
- Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(2), 239–256 (1992)
- Bulow, H., Birk, A.: Scale-free registration in 3-d: 7 degrees of freedom with fourier mellin soft transforms. *Int. J. Comp. Vision* **126**, 731–750 (2018)
- Bustos, A.P., Chin, T.J.: Guaranteed outlier removal for point cloud registration with correspondences. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(12), 2868–2882 (2018). <https://doi.org/10.1109/TPAMI.2017.2773482>
- Campbell, D., Petersson, L.: Gogma: Globally-optimal gaussian mixture alignment. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5685–5694 (2016)
- Chetverikov, D., Svirkov, D., Stepanov, D., Krsek, P.: The trimmed iterative closest point algorithm. In: Object recognition supported by user interaction for service robots, vol. 3, pp. 545–548 vol.3 (2002)
- Choi, S., Zhou, Q.Y., Miller, S., Koltun, V.: A large dataset of object scans. [arXiv:1602.02481](https://arxiv.org/abs/1602.02481) (2016)
- Choy, C., Dong, W., Koltun, V.: Deep global registration. In: CVPR (2020)
- Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 8957–8965 (2019). <https://doi.org/10.1109/ICCV.2019.00905>
- D. Evangelidis, G., Kounades-Bastian, D., Horaud, R., Psarakis, E.: A generative model for the joint registration of multiple point sets (2014)
- Darom, T., Keller, Y.: Scale-invariant features for 3-d mesh models. *IEEE Trans. Image Process.* **21**(5), 2758–2769 (2012)
- Efraim, A., Francos, J.M.: The universal manifold embedding for estimating rigid transformations of point clouds. In: 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5157–5161 (2019)
- Ferencz, I.H., Shimshoni, I.: Registration of 3d point clouds using mean shift clustering on rotations and translations. 2017 Int. Conf. 3D Vis. (3DV) pp. 374–382 (2017)
- Fischler, M., Bolles, R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graph. Image Process.* **24**, 381–395 (1981)
- Fuhrmann, S., Langguth, F., Gesele, M.: Mve: A multi-view reconstruction environment. In: Proceedings of the Eurographics Workshop on Graphics and Cultural Heritage, GCH '14, pp. 11–18. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland (2014)
- Garcia-Garcia, A., Gomez-Donoso, F., Garcia-Rodriguez, J., Orts-Escolano, S., Cazorla, M., Azorin-Lopez, J.: Pointnet: A 3d convolutional neural network for real-time object class recognition. In: 2016 International Joint Conference on Neural Networks (IJCNN), pp. 1578–1584 (2016)
- Gelfand, N., Mitra, N.J., Guibas, L.J., Pottmann, H.: Robust global registration. In: Symposium on geometry processing, vol. 2, p. 5. Vienna, Austria (2005)
- Gojcic, Z., Zhou, C., Wegner, J.D., Wieser, A.: The perfect match: 3d point cloud matching with smoothed densities. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5540–5549 (2019). <https://doi.org/10.1109/CVPR.2019.00569>
- Grant, W.S., Voorhies, R.C., Itti, L.: Finding planes in lidar point clouds for real-time registration. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4347–4354 (2013). <https://doi.org/10.1109/IROS.2013.6696980>
- Grigsby, S.: Leaf-on lidar point cloud data for solar site assessment of the cu-boulder campus. Department of Geography, University of Colorado at Boulder, digital media (2013)
- Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J., Kwok, N.M.: A comprehensive performance evaluation of 3d local feature descriptors. *Int. J. Comput. Vision* **116**(1), 66–89 (2016)
- Hackel, T., Wegner, J., Schindler, K.: Fast semantic segmentation of 3d point clouds with strongly varying density. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inform. Sci.* **III-3**, 177–184 (2016). <https://doi.org/10.5194/isprsannals-III-3-177-2016>
- Hagege, R., Francos, J.M.: Parametric estimation of affine transformations: an exact linear solution. *J. Math. Imag. Vis.* **37**(1), 1–16 (2010)
- Horn, B.K.P.: Extended gaussian images. *Proc. IEEE* **72**(12), 1671–1686 (1984)
- Horn, B.K.P.: Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. A* **4**(4), 629–642 (1987)
- Horn, B.K.P., Hilden, H., Negahdaripour, S.: Closed-form solution of absolute orientation using orthonormal matrices. *J. Opt. Soc. Am. A* **5**(7), 1127–1135 (1988)
- Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **21**(5), 433–449 (1999)
- Lee, T.k., Lim, S., Lee, S., An, S., Oh, S.y.: Indoor mapping using planes extracted from noisy rgb-d sensors. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1727–1733 (2012). <https://doi.org/10.1109/IROS.2012.6385909>
- Liu, X., Qi, C.R., Guibas, L.J.: Flownet3d: Learning scene flow in 3d point clouds. In: CVPR 2019 (2018)

34. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision (ijcai) (1981)
35. Maes, C., Fabry, T., Keustermans, J., Smeets, D., Suetens, P., Vandermeulen, D.: Feature detection on 3d face surfaces for pose normalisation and recognition. pp. 1–6 (2010)
36. Magnusson, M., Lilienthal, A., Duckett, T.: Scan registration for autonomous mining vehicles using 3d-ndt. *J. Field Robot.* **24**(10), 803–827 (2007). <https://doi.org/10.1002/rob.20204>
37. Makadia, A., Patterson, A., Daniilidis, K.: Fully automatic registration of 3d point clouds. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), vol. 1, pp. 1297–1304 (2006)
38. Mellado, N., Aiger, D., Mitra, N.J.: Super 4pcs fast global point-cloud registration via smart indexing. *Comput. Graph. Forum* **33**(5), 205–215 (2014)
39. Myronenko, A., Song, X.: Point set registration: coherent point drift. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(12), 2262–2275 (2010)
40. Myronenko, A., Song, X.B.: On the closed-form solution of the rotation matrix arising in computer vision problems. *ArXiv arXiv:abs/0904.1613* (2009)
41. Neely, A.: Quantifying rock strength controls in Guadalupe Mountains, NM/TX. National Center for Airborne Laser Mapping (NCALM), distributed by OpenTopography (2019)
42. OSU: B4 project - southern san andreas and san jacinto faults. Ohio State University, School of Earth Sciences with collaboration with National Center for Airborne Laser Mapping (NCALM), distributed by OpenTopography (2006)
43. Pathak, K., Birk, A., Vaškevičius, N., Poppinga, J.: Fast registration based on noisy planes with unknown correspondences for 3d mapping. *IEEE Trans. Rob.* **26**(3), 424–441 (2010). <https://doi.org/10.1109/TRO.2010.2042989>
44. Pottmann, H., Leopoldseder, S., Hofer, M.: Registration without icp. *Comput. Vis. Image Underst.* **95**(1), 54–71 (2004)
45. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. In: Proceedings Third International Conference on 3-D Digital Imaging and Modeling, pp. 145–152 (2001)
46. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: 2009 IEEE International Conference on Robotics and Automation, pp. 3212–3217 (2009)
47. Rusu, R.B., Blodow, N., Marton, Z.C., Beetz, M.: Aligning point cloud views using persistent feature histograms. In: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3384–3391 (2008)
48. Sipiran, I., Bustos, B.: Harris 3d: a robust extension of the harris operator for interest point detection on 3d meshes. *Vis. Comput.* **27**, 963–976 (2011)
49. Slater, L.J.E.: Lincoln national forest, NM: Floodplains in Fluvial Networks. National Center for Airborne Laser Mapping (NCALM), distributed by OpenTopography (2012)
50. Tam, G.K.L., Cheng, Z., Lai, Y., Langbein, F.C., Liu, Y., Marshall, D., Martin, R.R., Sun, X., Rosin, P.L.: Registration of 3d point clouds and meshes: a survey from rigid to nonrigid. *IEEE Trans. Visual Comput. Graph.* **19**(7), 1199–1217 (2013)
51. Vaskevicius, N., Birk, A., Pathak, K., Schwertfeger, S.: Efficient representation in 3d environment modeling for planetary robotic exploration. *Adv. Robot.* **24** (2010)
52. Wang, Y., Solomon, J.: Deep closest point: Learning representations for point cloud registration. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 3522–3531 (2019)
53. Wasklewicz, T.: Death valley national park, ca: Badwater basin. National Center for Airborne Laser Mapping (NCALM), distributed by OpenTopography (2013)
54. Yang, H., Shi, J., Carlone, L.: Teaser: fast and certifiable point cloud registration. *IEEE Trans. Rob.* **37**(2), 314–333 (2021). <https://doi.org/10.1109/TRO.2020.3033695>
55. Yang, J., Cao, Z., Zhang, Q.: A fast and robust local descriptor for 3d point cloud registration. *Inf. Sci.* **346–347**, 163–179 (2016)
56. Yang, J., Li, H., Campbell, D., Jia, Y.: Go-icp: a globally optimal solution to 3d icp point-set registration. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(11), 2241–2254 (2016)
57. Yang, J., Xiao, Y., Cao, Z.: Toward the repeatability and robustness of the local reference frame for 3d shape matching: an evaluation. *IEEE Trans. Image Process.* **27**(8), 3766–3781 (2018)
58. Yang, J., Zhang, Q., Xian, K., Xiao, Y., Cao, Z.: Rotational contour signatures for both real-valued and binary feature representations of 3d local shape. *Comput. Vis. Image Underst.* **160**, 133–147 (2017)
59. Yuan, W., Eckart, B., Kim, K., Jampani, V., Fox, D., Kautz, J.: Deepgmr: Learning latent gaussian mixture models for registration. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) *Computer Vision - ECCV 2020*, pp. 733–750. Springer International Publishing, Cham (2020)
60. Zabulis, X., Lourakis, M.I.A., Koutlemanis, P.: Correspondence-free pose estimation for 3d objects from noisy depth data. *Vis. Comput.* **34**(2), 193–211 (2018)
61. Zhang, Z.: Iterative point matching for registration of free-form curves and surfaces. *Int. J. Comput. Vision* **13**(2), 119–152 (1994)
62. Zhou, Q.Y., Park, J., Koltun, V.: Fast global registration. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *Computer Vision - ECCV 2016*, pp. 766–782. Springer International Publishing, Cham (2016)
63. Zhou, Q.Y., Park, J., Koltun, V.: Open3D: A modern library for 3D data processing. *arXiv:1801.09847* (2018)

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



Amit Efraim received the B.Sc and M.Sc degrees in electrical and computer engineering in 2010 and 2017, respectively, from Ben-Gurion University, Beer Sheva, Israel. He is currently a Ph.D. student at the Electrical and Computer Engineering Department, Ben-Gurion University. His research interests are in image and point cloud registration and in geometric problems in computer vision.



Joseph M. Francos received the B.Sc. degree in computer engineering in 1982, and the D.Sc. degree in electrical engineering in 1991, both from the Technion-Israel Institute of Technology, Haifa. From 1991 to 1992, he was with the Department of Electrical Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, as a Visiting Assistant Professor. In 1993, he joined the Department of Electrical and Computer Engineering, Ben-Gurion University, Beer-Sheva, Israel, where

he is now a Professor. He heads the Mathematical Imaging Group, and the Signal Processing track. He also held visiting positions at the Massachusetts Institute of Technology Media Laboratory, Cambridge, at the Electrical and Computer Engineering Department, University of California, Davis, at the Electrical Engineering and Computer Science Department, University of Illinois, Chicago, INRIA Sophia-Antipolis, France, and at the Electrical Engineering Department, University of California, Santa Cruz. His current research interests are in parametric modeling and estimation of multi-dimensional signals, image registration, estimation of object deformations from images, manifold learning, parametric modeling and estimation of 2D random fields, random fields theory.