



Robust affine registration method using line/surface normals and correntropy criterion

Abdurrahman Yilmaz¹ · Hakan Temeltas¹

Received: 17 August 2021 / Accepted: 26 November 2021 / Published online: 4 January 2022
© The Author(s) 2021

Abstract

The problem of matching point clouds is an efficient way of registration, which is significant for many research fields including computer vision, machine learning, and robotics. There may be linear or non-linear transformation between point clouds, but determining the affine relation is more challenging among linear cases. Various methods have been presented to overcome this problem in the literature and one of them is the affine variant of the iterative closest point (ICP) algorithm. However, traditional affine ICP variants are highly sensitive to effects such as noises, deformations, and outliers; the least-square metric is substituted with the correntropy criterion to increase the robustness of ICPs to such effects. Correntropy-based robust affine ICPs available in the literature use point-to-point metric to estimate transformation between point clouds. Conversely, in this study, a line/surface normal that examines point-to-curve or point-to-plane distances is employed together with the correntropy criterion for affine point cloud registration problems. First, the maximum correntropy criterion measure is built for line/surface normal conditions. Then, the closed-form solution that maximizes the similarity between point sets is achieved for 2D registration and extended for 3D registration. Finally, the application procedure of the developed robust affine ICP method is given and its registration performance is examined through extensive experiments on 2D and 3D point sets. The results achieved highlight that our method can align point clouds more robustly and precisely than the state-of-the-art methods in the literature, while the registration time of the process remains at reasonable levels.

Keywords Linear registration · Affine iterative closest point · Correntropy · 2D and 3D point clouds

Introduction

Today, point set/cloud registration is a noteworthy problem since it contributes to various fields such as computer vision, pattern recognition, image processing and classification (see in [1–4]), medical imaging and diagnosis (see in [5]), 3D modeling and construction (see in [6–8]), machine learning (see in [9]) and other engineering fields. The point cloud registration problem can be defined as finding the spatial transformation between two point sets by aligning them in a space. Furthermore, with the proliferation of 3D LIDARS,

point cloud registration has gained importance for mapping and localization problems in robotics (see [10–12] studies).

As explained in [13], point cloud registration can be categorized into optimization-based and learning-based registration methods. Currently, learning-based registration techniques are very popular, for example, a global registration method based on deep-learning has been developed in [14], and a deep outlier elimination method for the correspondence determination/rejection stage of registration via spatial consistency has also been presented in [15]. In addition, Huang et al. tried to solve the point cloud registration problem with low overlap ratio using learning techniques [16], and RGB color and depth information was also considered in the feature extraction process of registration by Wan et al. [17]. However, the main drawbacks of such approaches are the need for large training sets, unexpected performance loss in unknown scenes, and no guarantees of similar performance outside the training area [13].

The iterative closest point (ICP) algorithm, which is one of the most common optimization-based registration

✉ Abdurrahman Yilmaz
yilmazabdurrah@itu.edu.tr

Hakan Temeltas
temeltash@itu.edu.tr

¹ Control and Automation Engineering Department, Electrical and Electronics Engineering Faculty, Istanbul Technical University, Ayazaga Campus, Maslak, 34469 Sarýyer/Istanbul, Turkey

approaches, can be generalized and guarantee convergence owing to its powerful mathematical infrastructure. For this reason, ICP-like approaches are used together with learning-based approaches during the refinement stage to improve registration performance. For example, it has been observed by [14] that the registration accuracy has increased approximately seven times with ICP refinement, which once again reveals the importance of optimization-based registration approaches.

From a different perspective [18], the registration problem can be roughly divided into two main sub-branches in terms of transformation types; these are linear and non-linear registration. Rigid registration methods, a subset of linear approaches, only have satisfactory registration performance under combinations of rigid motions such as translation, reflection, and rotation movements. On the other hand, affine registration methods can both solve rigid registration problems and overcome the shear mapping and scaling effects.

The rigid ICP, which aligns 3D shapes to estimate the relation between them in terms of a rotation matrix and a translation vector, is developed by [19]. Several enhanced variants of the traditional ICP method have been presented, including point-to-line ICP by [20], generalized ICP by [21], globally optimal ICP by [22], and other variants available in the review study by [23]. To make ICP suitable for linear but not rigid registration problems, the ICP variants such as scale ICP by [24,25] and affine ICP by [26,27] have been proposed. However, these ICP variants are not robust to noises, outliers, and/or deformations (see [28,29] studies) since the least-square criterion is used to construct objective functions to be optimized.

It is obvious that such effects (noises, outliers, etc.) are a fact of real-world applications, and successful and accurate registration under these effects is very important. That is why, robust ICP algorithms including correntropy-based robust ICP by [30], bidirectional correntropy-based robust ICP by [31], probabilistic ICP by [32], robust scale ICP by [33–35], robust affine ICP by [29] are proposed for linear registration, on the other hand, a cluster correspondence-based fast method by [18], expectation-maximization and/or Gaussian mixture model(GMM)-based robust methods by [36–38], coherent point drift(CPD)-based robust approaches by [39,40] are developed for non-linear registration.

In studies where variants of the linear registration problem such as rigid, scaled rigid, and affine registration are solved with the use of the correntropy criterion, it is observed that the robustness and matching performance are quite well. Correntropy is a similarity measure between two random variables and by considering model and data sets as random variables, it is possible to eliminate outliers and/or noises better and to find the transformation between the point clouds more precisely.

In this study, we propose a robust affine registration method that combines line normals in 2D space, surface normals in 3D space, and correntropy criterion. The proposed registration algorithm is novel since other robust ICPs available in the literature, which employ correntropy criteria for linear registration problems, utilize point-to-point metrics to construct objective functions. The closed-form solution of the presented approach is achieved for both 2D and 3D registration problems. The extensive experiments show the superiority of the developed method in terms of accuracy rate and efficiency over the CPD method by [39], traditional ICP algorithms developed for affine registration by [26], and robust affine ICP approach by [29]. Although the performance and matching precision increase with the developed method, the elapsed time for each registration stays at tolerable levels.

The rest of this study is organized as follows: in the next section, preliminary information on the linear registration problem is given in detail. The proposed robust affine registration method based on line/surface normal and correntropy criterion is explained in the third section. The experiments are conducted on registration of 2D and 3D well-known shapes and the results achieved that validate the accuracy, robustness, and efficiency of the method developed are shown in the fourth section. The penultimate section includes some discussions and comments on the method developed and the results achieved. The concluding remarks and future work suggestions are stated in the last section.

Overview of linear registration

In the registration problem using feature-based similarity measure given in [41] as

$$\min_{j(i) \in \{1, 2, \dots, N_m\}} \sum_{i=1}^{N_p} \|\mathcal{T}(\mathbf{p}_i) - \mathbf{m}_{j(i)}\|^2, \quad (1)$$

$\mathcal{T}(\cdot)$ term represents the linear or non-linear transformation between the n -dimensional point clouds to be registered defined as model set $M \triangleq \{\mathbf{m}_i\}_{i=1}^{N_m}$ and data set $P \triangleq \{\mathbf{p}_i\}_{i=1}^{N_p}$. To construct the objective function, the corresponding point of \mathbf{p}_i , the i^{th} element of the data set, in model set $\mathbf{m}_{j(i)}$ is utilized.

Rigid registration

In the simplest form of linear registration, as described in [30], the expression in (1) turns into

$$\min_{\mathbf{R}, \mathbf{t}, j(i) \in \{1, 2, \dots, N_m\}} \sum_{i=1}^{N_p} \|\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{m}_{j(i)}\|^2 \quad (2)$$

and this is called as rigid registration. The transformations between point clouds including only combinations of rigid motions such as rotation, translation, and reflection can be determined by utilizing many methods available in the literature, but traditional ICP developed by [19] is one of the most preferred methods. By minimizing the objective function in (2), traditional ICP computes the transformation between point clouds in terms of a rotation matrix (\mathbf{R}) and a translation vector (\mathbf{t}).

ICP for affine registration

The ICP variant, which aims to express linear but not rigid transformation between point clouds by an affine relation, was proposed by [26] as

$$\min_{\mathbf{A}, \mathbf{t}, j(i) \in \{1, 2, \dots, N_m\}} \sum_{i=1}^{N_p} \|(\mathbf{A}\mathbf{p}_i + \mathbf{t}) - \mathbf{m}_{j(i)}\|^2. \tag{3}$$

Here, affine ICP uses an affine matrix (\mathbf{A}) and translation vector (\mathbf{t}) to describe the current transformation between point clouds and this method can even handle scaling and skew (shear) effects.

Least-square and correntropy criteria

In general, ICP algorithms consider the point cloud registration problem as a least-square minimization problem. Using this idea, the difference between the transformed data set and the model set is tried to be minimized. Another approach, utilization of correntropy criterion, aims to find the transformation that maximizes the similarity between point sets. Correntropy combines correlation term in probability and entropy term in mechanics and is defined as a similarity measure between two random variables \mathcal{X} and \mathcal{Y} . The mathematical description of the correntropy is given in [42] as

$$\mathbf{V}_\sigma(\mathcal{X}, \mathcal{Y}) = \mathbf{E}[\kappa_\sigma(\mathcal{X} - \mathcal{Y})] \tag{4}$$

where $\mathbf{E}[\cdot]$ denotes expectation in probability and $\kappa_\sigma(\cdot)$ represents a symmetric positive definite kernel with σ kernel width. For the Gaussian Kernel, Wu et al. [42] is achieved the correntropy definition as

$$\mathbf{V}_\sigma(\mathcal{X}, \mathcal{Y}) = \mathbf{E}\left[\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\mathcal{X} - \mathcal{Y})^2}{2\sigma^2}\right)\right]. \tag{5}$$

Affine ICP with correntropy criterion

The definition of correntropy criterion in (5) was used by [29] to generate a robust affine ICP variant. The new objective function is constructed as

$$\max_{\substack{\det(\mathbf{A}) \neq 0, \mathbf{t}, \\ j(i) \in \{1, 2, \dots, N_m\}}} \sum_{i=1}^{N_p} \exp\left(-\frac{\|(\mathbf{A}\mathbf{p}_i + \mathbf{t}) - \mathbf{m}_{j(i)}\|^2}{2\sigma^2}\right) \tag{6}$$

where the transformation between the model and data sets is computed through maximum correntropy criterion instead of least-square-based minimization. In the same study (see [29] study), the researchers showed that ICP variants using the correntropy criterion perform better than those utilizing the least-square criterion, especially when the point clouds contain noises and/or outliers. That is why, the robust affine point cloud registration method, which we propose in this study, uses the correntropy criterion.

Affine ICP with line/surface normals and correntropy criterion

In all methods mentioned so far, the point-to-point metric is used while deriving the cost function, which takes into consideration the distance from one point to another (see in Fig. 1a). With line/surface normals, it is also possible to derive methods that consider the distance from a point to the defined line or plane (see in Fig. 1b, c). [20] proposed a variant of traditional ICP that uses point-to-line metric to match 2D point sets. However, affine ICP variant using the correntropy criterion and line/surface normals is not available in the literature. As a result, this improved version of affine ICP is introduced in this study to solve the affine registration problem more robustly for both 2D and 3D point clouds.

The objective function using line/surface normals and correntropy criterion can be constructed as

$$J_{PL}^{corr}(\mathbf{A}, \mathbf{t}) = \sum_{i=1}^{N_p} \exp\left(-\frac{(\mathbf{n}_i^T \cdot [(\mathbf{A}\mathbf{p}_i + \mathbf{t}) - \mathbf{m}_{j(i,1)}])^2}{2\sigma^2}\right) \tag{7}$$

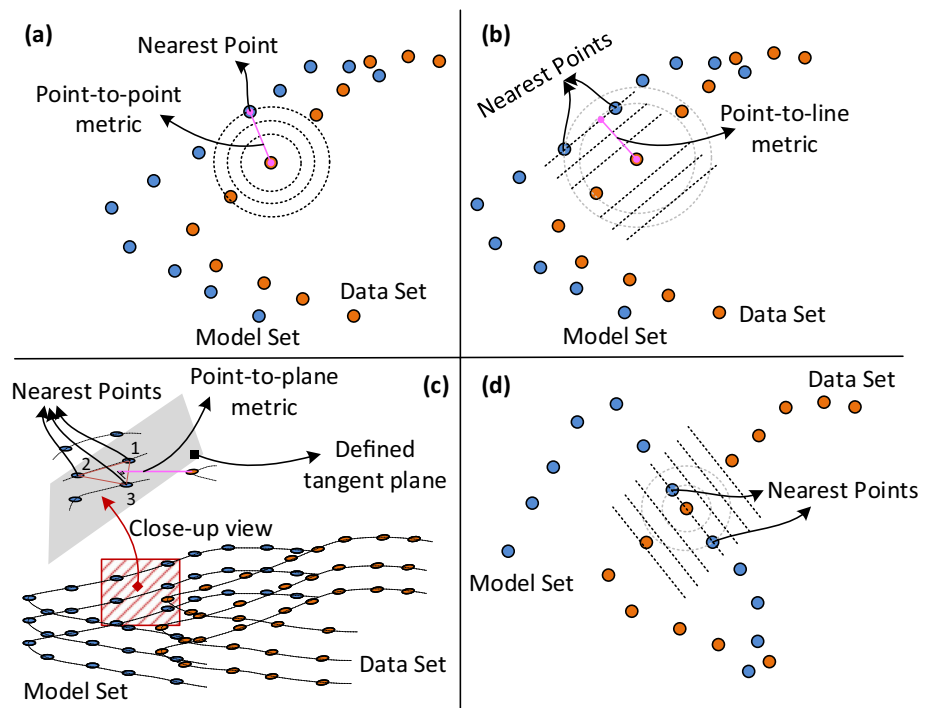
where \mathbf{n}_i is i^{th} normal vector and $\mathbf{m}_{j(i,1)}$ first correspondence of \mathbf{p}_i in the model set. The superscript ‘*corr*’ represents the correntropy criterion and ‘*PL*’ stands for **P**oint-to-**L**ine or **P**oint-to-**P**lane metrics utilized based on dimension of point clouds to be registered.

Computation of affine transformation

ICP algorithms find closest points iteratively and for the k^{th} iteration of an ICP variant two steps given as follows are solved:

1. **Finding corresponding points:** To define a tangent line in 2D space, at least two nearest points of \mathbf{p}_i , first and second correspondences ($\mathbf{m}_{j(i,1)}$ and $\mathbf{m}_{j(i,2)}$), are required.

Fig. 1 Visual demonstrations of (a) point-to-point, (b) point-to-line, (c) point-to-plane metrics and (d) not desired configuration for point-to-line metric case



On the other hand, three corresponding points ($\mathbf{m}_{j(i,1)}$, $\mathbf{m}_{j(i,2)}$ and $\mathbf{m}_{j(i,3)}$) are necessary to find the tangent plane of a surface in 3D space. These correspondences can be determined using the previous transformation (\mathbf{A}_{k-1} , \mathbf{t}_{k-1}) via many methods available in the literature such as *Delaunay tessellation* of [43], *k-d trees* of [44], *probabilistic correspondence* method of [5]. In this study, the *k-d trees* approach is applied to assign corresponding points of \mathbf{p}_i in the model set.

2. **Computing transformation:** The affine transformation maximizing the similarity between point clouds in k^{th} iteration, (\mathbf{A}_k , \mathbf{t}_k), can be calculated by maximizing the objective function given in (7). The derivation-based solution procedure of this optimization problem will be explained further.

This iterative process continues until it reaches the predefined maximum number of iterations, or stops if the difference between the current and last transformations remains within a sufficiently small boundary.

Computation procedure for 2D point clouds

In an affine transformation ($\mathbf{A}_{n \times n}$, $\mathbf{t}_{n \times 1}$), the subscripts of matrix \mathbf{A} and vector \mathbf{t} represent the dimensions of them and are related to the dimension of point clouds (n). In the 2D workspace (for $n = 2$), the unknown parameters in \mathbf{A} and \mathbf{t} can be gathered in a vector $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5, x_6] \triangleq [\mathbf{a}_{11}, \mathbf{a}_{12}, \mathbf{a}_{21}, \mathbf{a}_{22}, \mathbf{t}_x, \mathbf{t}_y]$ where \mathbf{a}_{ij} is the elements of matrix \mathbf{A} at the i^{th} row and j^{th} column. With the usage of \mathbb{P}_i definition

$$\mathbb{P}_i = \begin{bmatrix} \mathbf{p}_i^T & \mathbf{0} & 1 & 0 \\ \mathbf{0} & \mathbf{p}_i^T & 0 & 1 \end{bmatrix}$$

and vector \mathbf{x} , the objective function in (7) can be rewritten as

$$J(\mathbf{x}) = \sum_{i=1}^{N_p} \exp \left(- \frac{(\mathbb{P}_i \mathbf{x} - \pi_i)^T C_i (\mathbb{P}_i \mathbf{x} - \pi_i)}{2\sigma^2} \right) \tag{8}$$

where C_i is equal to $\mathbf{n}_i \cdot \mathbf{n}_i^T$, and π_i stands for $\mathbf{m}_{j(i,1)}$. To calculate optimal \mathbf{x} that maximizes the objective function in (8), derivation-based maximization methods can be utilized such that $\frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} = 0 \rightarrow \mathbf{x} = \mathbf{x}^*$. If this derivation is applied on the extended version of the objective function as follows:

$$J(\mathbf{x}) = \sum_{i=1}^{N_p} \exp \left(-\frac{\mathbf{x}^T \mathbb{P}_i^T C_i \mathbb{P}_i \mathbf{x} + \pi_i^T C_i \pi_i - 2\pi_i^T C_i \mathbb{P}_i \mathbf{x}}{2\sigma^2} \right),$$

the expression

$$\sum_{i=1}^{N_p} \left[\mathbf{x}^T \left(\mathbb{P}_i^T C_i \mathbb{P}_i + \left(\mathbb{P}_i^T C_i \mathbb{P}_i \right)^T \right) - 2\pi_i^T C_i \mathbb{P}_i \right] \Delta_i = 0 \tag{9}$$

is obtained in a compact form. Here, Δ_i corresponds to the exponential part of the objective function which is $\exp \left(-\frac{\mathbf{x}^T \mathbb{P}_i^T C_i \mathbb{P}_i \mathbf{x} + \pi_i^T C_i \pi_i - 2\pi_i^T C_i \mathbb{P}_i \mathbf{x}}{2\sigma^2} \right)$. The expression in (9) can be arranged as

$$\sum_{i=1}^{N_p} \left(\mathbf{x}^T \mathbb{P}_i^T C_i \mathbb{P}_i - \pi_i^T C_i \mathbb{P}_i \right) \Delta_i = 0 \tag{10}$$

since C_i is a symmetric matrix. As a result, using $\mathbf{H} \triangleq \sum_{i=1}^{N_p} \mathbb{P}_i^T C_i \mathbb{P}_i \Delta_i$ and $\mathbf{g} \triangleq \sum_{i=1}^{N_p} \pi_i^T C_i \mathbb{P}_i \Delta_i$ definitions, the optimal \mathbf{x} is achieved as

$$\mathbf{x}^* = \left(\mathbf{g} \cdot \mathbf{H}^{-1} \right)^T. \tag{11}$$

Here, the necessary terms to construct affine transformation matrix (\mathbf{A}, \mathbf{t}) that maximizes the similarity between the point clouds (model and data sets) are calculated with an assumption that matrix \mathbf{H} is an invertible matrix.

The assumption about the non-singularity of matrix \mathbf{H} is valid except for the cases visually demonstrated in Fig. 1d. This means that the relevant point in the data set is also lying on the line/plane defined using the model set. The points satisfying this condition should be ignored while constructing the objective function to eliminate singularity.

On the other hand, it is obvious that the terms \mathbf{g} and \mathbf{H} used to derive optimal \mathbf{x} also include the term \mathbf{x} in the term Δ , and this indicates that there is an algebraic loop in the solution. However, since ICP is an iterative approach the optimal \mathbf{x} vector can be computed. In each iteration, Δ is computed using the previous transformation $(\mathbf{A}_{k-1}, \mathbf{t}_{k-1})$ to achieve $(\mathbf{A}_k, \mathbf{t}_k)$.

Extension to 3D point cloud registration cases

The computation procedure of affine transformation is completely the same for 2D and 3D registration problems. The only difference is that the dimensions of the matrices and vectors such as $\mathbf{x}, \mathbf{n}_i, \mathbb{P}_i$ change with respect to the dimensions of the point clouds to be registered. Similar to the 2D case, unknown parameters vector can be constructed as $\mathbf{x} \triangleq [\mathbf{a}_{11}, \mathbf{a}_{12}, \dots, \mathbf{a}_{33}, \mathbf{t}_x, \mathbf{t}_y, \mathbf{t}_z]$ for 3D space. Instead of a tangent line of a curve in 2D, \mathbf{n}_i is defined using the tangent plane of a surface. The dimension of \mathbb{P}_i is also extended such that

$$\mathbb{P}_i^{3D} = \begin{bmatrix} \mathbf{p}_i^T & \mathbf{0} & \mathbf{0} & 1 & 0 & 0 \\ \mathbf{0} & \mathbf{p}_i^T & \mathbf{0} & 0 & 1 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{p}_i^T & 0 & 0 & 1 \end{bmatrix}$$

is written as new \mathbb{P}_i for 3D space.

In conclusion, the mathematical expression of the proposed affine registration method and a detailed explanation of the solution procedure are given up to here. In the next section, studies on verification of the proposed method in terms of robustness, success rate, accuracy, etc. will be discussed.

Experimental results

To verify the algorithm proposed in this study in terms of accuracy and robustness, its registration performance is investigated on well-known 2D and 3D shapes. In the following sub-sections, the applied procedure to practice matching experiments is given and the superiority of the developed algorithm over other state-of-the-art algorithms is shown in detail.

AfICP-PL-corr for 2D registration

2D registration experiments are conducted on eight different 2D shapes. In each experiment, the edges of the 2D image are extracted to generate the model set. A model set extraction example for the bat experiment is demonstrated in Fig. 2. The extracted shapes are utilized to construct data set and six different model sets. An example of data and model sets for apple experiments is shown in Fig. 3. Here, to achieve the shape in Fig. 3b, the shape in Fig. 3a which is called the origi-

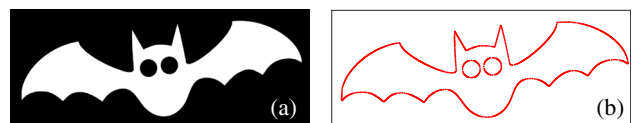


Fig. 2 Extract model set from well-known images, bat example: (a) bat image, (b) model set from the image by extracting the edges.

Fig. 3 Data and model sets before registration: apple example, (a) original model set (MS), (b) data set, (c) cropped MS, (d) MS with uniform noise, (e) MS with Gaussian noise, (f) cropped MS with uniform noise, and (g) cropped MS with Gaussian noise

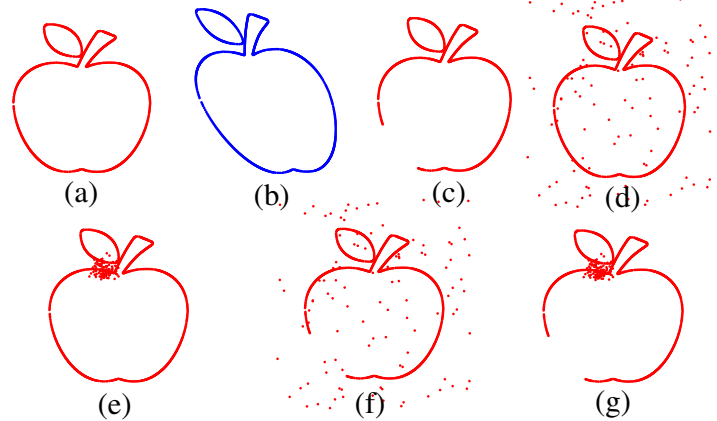


Table 1 The number of points in the 2D shapes, and the upper and lower boundaries of the 2D shapes registered in this study

Shape	Quantity	(y, \bar{y})	(x, \bar{x})
Apple	1061	(−390, 270)	(−250, 270)
Bat	1572	(−180, 190)	(−470, 490)
Bunny	982	(−260, 310)	(−240, 280)
Butterfly	1000	(−750, 640)	(−1000, 950)
Dog	981	(−230, 260)	(−350, 270)
Dragon	1178	(−810, 740)	(−860, 910)
Fish	1046	(−1260, 1520)	(−3040, 3180)
Leaf	1320	(−280, 240)	(−240, 230)

nal model set is transformed with an affine relation including rotation, translation, shearing, and scaling effects. The left bottom side of the original model set is deleted to examine the missing data effect on registration and the cropped model set is achieved as in Fig. 3c. A uniform noise is added to the original and cropped model sets to reach the model sets in Fig. 3d, f, respectively. Similarly, the model sets in Fig. 3e, g are formed by adding a Gaussian noise on original and cropped model sets, respectively.

Table 1 shows the structural properties of the 2D point sets generated. The names of the shapes are given in the first column and the number of points in the original 2D shapes is placed in the second column. To generate noisy point sets, 20% of the number of points in the original point set is added based on relevant noise distribution. Similarly, the left bottom of the shapes (about one-eighth) is cropped to construct model sets having missing data. The upper and lower boundaries of the original model set of the shapes are given for y and x -axes in the third and fourth columns, respectively.

The performance of our algorithm is examined on all model set variants in Fig. 3. The achieved results are compared to traditional affine ICP (*AfICP-PP-ls*) by [26] and affine ICP with point-to-line metric (*AfICP-PL-ls*) algorithms to see the superiority of our algorithm under noises

and outliers. The experiments are conducted by MATLAB 2019a® on a PC with Intel Core i7 Q740 1.73 GHz CPU and 4 GB RAM.

When the model set has no outliers, noises, and/or missing data (original model set), the model and data sets are matched successfully with all algorithms (*AfICP-PP-ls*, *AfICP-PL-ls*, and *AfICP-PL-corr*). However, when changes such as noise insertion and/or cropping (to derive model set having missing data) are applied on the model set, it is seen that our algorithm developed in this study is much more successful than *AfICP-PP-ls* and *AfICP-PL-ls* in registration. Two separate experiments showing this for the model sets having missing data and uniform noise and the model sets having missing data and Gaussian noise are given in Fig. 4a, b, respectively.

The results obtained so far are sufficient to show the success of our algorithm, but we made even greater changes on the point sets to force our algorithm further. Changes in point sets are clearly seen in the visuals to be given later on where matching performances of the algorithms are evaluated, but it may be necessary to mention verbally here too. Apple_2 and Apple_3 point sets were obtained by adding a bite to the apple shape and removing the leaf on the apple shape, respectively. A new butterfly shape (Butterfly_2) was derived by removing the antennae of the butterfly and by adding spots on its wings. Similarly, the dog shape with the tail down is modified to tail up to generate a new dog shape (Dog_2). A new dragon shape (Dragon_2) was obtained by softening the neck and chin of the previous dragon shape. On the other hand, the stem of the leaf shape has been removed to construct Leaf_2. While making changes to the tail and ears of the bunny shape to get Bunny Stand_2, half of the bat shape was clipped for Bat_2. As seen in Fig. 5a, even no noise insertion and/or part cropping is applied on the model set, it is seen that other algorithms fail to register the point sets, but our algorithm successfully matches the model and data sets despite these modifications.

The matching performances of algorithms on modified point sets having missing data and noises are also investi-

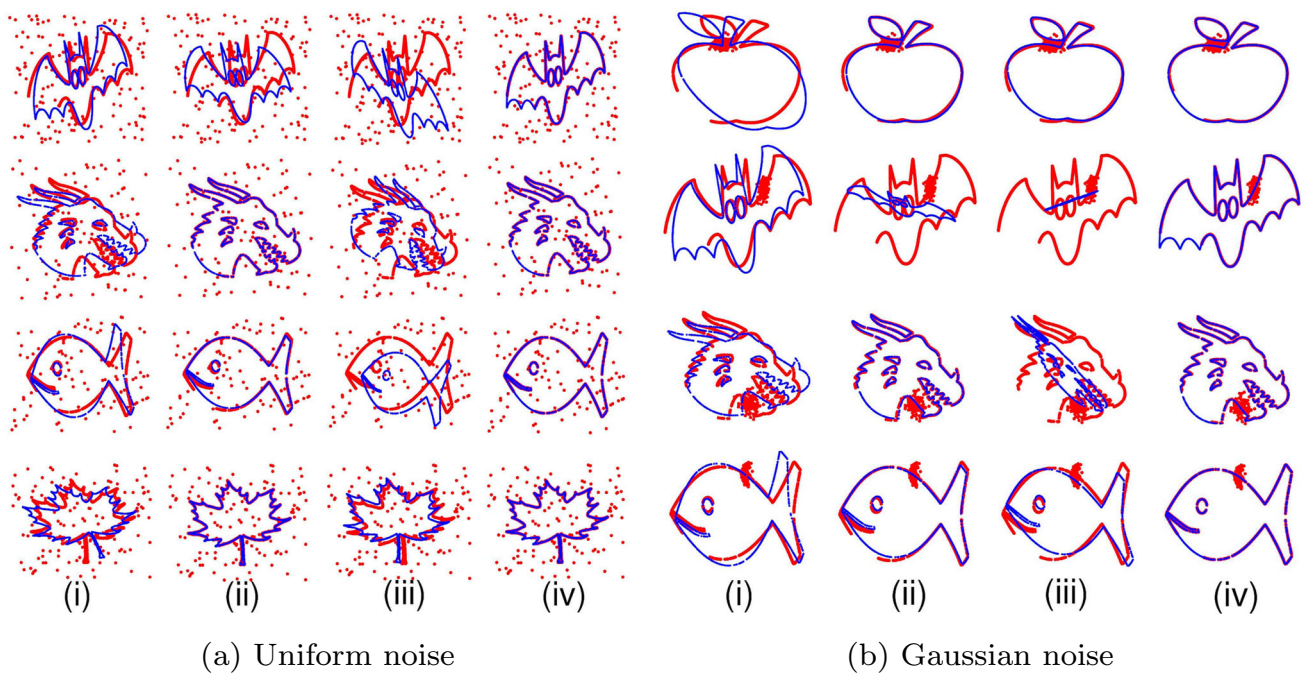


Fig. 4 Registration results of 2D shapes for model set having missing data and (a) uniform and (b) Gaussian noise: (i) the shapes before registration, registration results of (ii) *AfICP-PP-ls*, (iii) *AfICP-PL-ls* and (iv) *AfICP-PL-corr* (our algorithm)

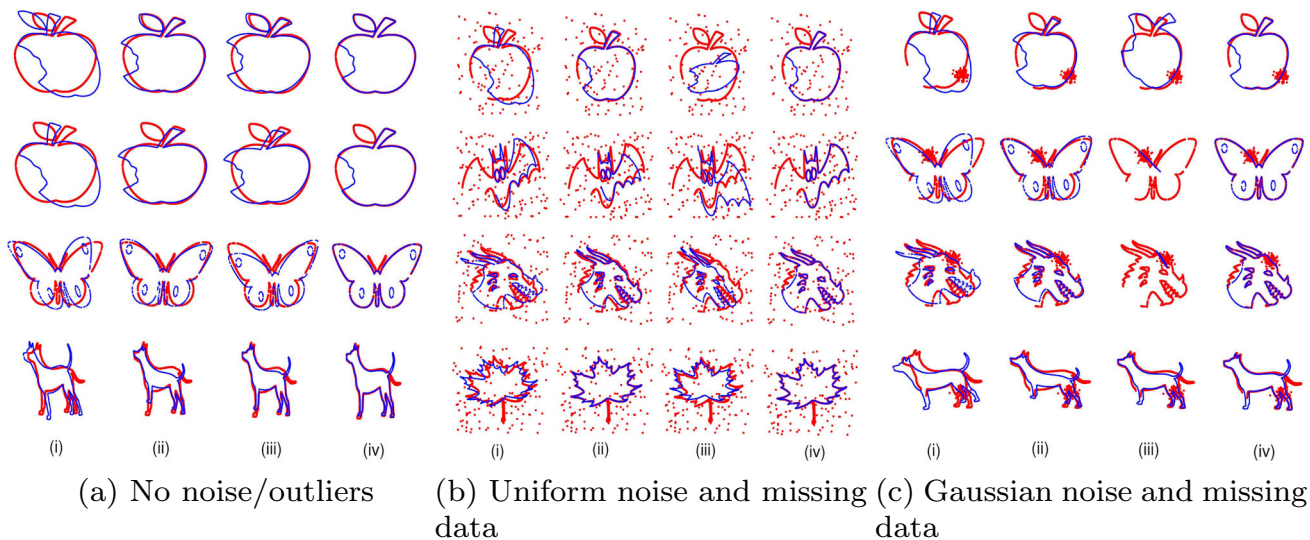


Fig. 5 Registration results of 2D shapes for modified model sets: (i) the shapes before registration, registration results of (ii) *AfICP-PP-ls*, (iii) *AfICP-PL-ls* and (iv) *AfICP-PL-corr* (our algorithm)

gated, and it is seen in Fig. 5b, c that the algorithm developed in this study registers successfully even under these disturbing effects.

The correntropy criterion-based method developed in this study (*AfICP-PL-corr*) so far has only been compared to conventional least-square criterion-based methods. However, Fig. 6 shows the comparison of *AfICP-PP-corr* (affine ICP with point-to-point metric and correntropy criterion) developed by [42] with our method *AfICP-PL-corr* in terms of

registration performance. The results obtained demonstrate that both algorithms are not affected by undesired factors such as noises, missing data, and/or outliers, and they are able to match the point sets successfully since their objective functions are constructed based on the correntropy criterion. However, in addition to the accuracy of the algorithms finding the transformation between two point sets, if we focus on how long it takes to find the correct transformation, it is seen that our algorithm (*AfICP-PL-corr*) requires much

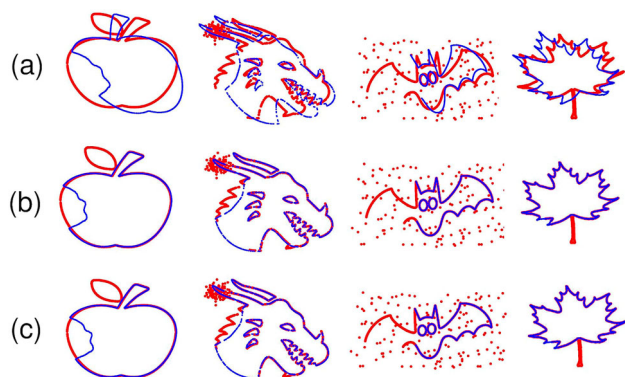


Fig. 6 Registration results of 2D shapes: various model set types—(a) the shapes before registration, registration results of (b) *AflCP-PL-corr* (our algorithm), (c) *AflCP-PP-corr*

fewer iterations to achieve the final transformation. An example indicating this is given in Fig. 7. The first column shows the registration of apple shape having no major changes, and after a while, all the algorithms have successfully calculated the correct transformation between the point sets. On the other hand, the bitten apple shape registration result is given in the second column and traditional least-squared criterion-based methods have failed to register. In addition to being successful, our algorithm does this with much less iteration than *AflCP-PP-corr*.

AflCP-PL-corr for 3D registration

Similar to 2D registration tests, well-known 3D shapes are taken into consideration to evaluate 3D affine registration performance of the algorithms. Glira's 3D point cloud data sets including Dino, Lion, etc. shapes have been mostly utilized in our applications (see [45] study).

Since 3D point clouds contain a lot of data, it is aimed to reduce the number of data in 3D point clouds by down-sampling them without disturbing the general features of the 3D shapes to make registration faster with algorithms such as ICP. Down-sampling methods are not in the scope of this study, but the available 3D down-sampling methods in MATLAB software are directly used to down-sample our 3D point clouds. An example of down-sampling on the 3D watering-can shape can be seen in Fig. 8.

The structural properties of the 3D point sets to be used for evaluating the 3D affine matching performances of the algorithms are listed in Table 2. The names of the shapes are given in the first column and the number of points in the original down-sampled 3D shapes registered is placed in the second column. To generate noisy point sets, 10% of the number of points in the original point set is added based on relevant noise distribution. Similarly, the left-front-bottom of the shapes (about one-eighth) is cropped to construct model sets having missing data. The upper and lower boundaries of

the original model set of the 3D shapes are given for z-, y-, and x-axes in the third, fourth, and fifth columns, respectively.

Similar to 2D cases, for each 3D registration experiment, the down-sampled 3D shapes are utilized to construct data set and six different model sets. An example of data and model sets is shown in Fig. 9. Here, the shape in Fig. 9a is called the original model set and this is transformed with an affine relation to achieve the data set in Fig. 9b. The left-front-bottom side of the original model set is erased to achieve the cropped model set in Fig. 9c. A uniform noise is inserted into original and cropped model sets to reach the model sets in Fig. 9d, f, respectively. Similarly, the model sets in Fig. 9e, g are created by adding a Gaussian noise on original and cropped model sets, respectively.

In 3D cases, the registration performance of the CPD algorithm was also considered for comparisons. Thus, it has been shown that the method proposed in this study is superior not only to other ICP-based variants but also to different approaches in the literature. CPD approach is an efficient registration method for point sets including noises and/or outliers since it converts the registration problem into probability density estimation by expressing point clouds in terms of the GMM.

All algorithms (*CPD*, *AflCP-PP-ls*, *AflCP-PL-ls* and *AflCP-PL-corr*) can find the correct transformation between model and data sets when there are no disturbing effects such as noise, missing data, etc. on the 3D model set. On the other hand, it is seen that our algorithm is much more successful in registration than *CPD*, *AflCP-PP-ls* and *AflCP-PL-ls* when changes such as noise addition and/or cropping (to generate a 3D model set having missing data) are applied to the model set. The affine 3D registration results of the algorithms, for example, are shown for model sets having missing data and uniform noise in Fig. 10 and for model sets having missing data and Gaussian noise in Fig. 11. Parts that are not correctly matched by *CPD*, *AflCP-PP-ls* and *AflCP-PL-ls* are indicated with arrows for better visualization and perception.

To force the algorithms further, we tested the partial registration performance of them on lion shape-based 3D point clouds in Fig. 12. As given in Fig. 13, even no noise insertion and/or part cropping is applied on the model set, it is seen that other algorithms fail to register 3D point clouds, but our algorithm successfully matches the model and data sets despite these major changes.

Numerical results for 2D/3D registration experiments

In addition to visual verification, the error metrics $\varepsilon_A = \|\mathbf{A}^* - \mathbf{A}\|_2$ and $\varepsilon_t = \|\mathbf{t}^* - \mathbf{t}\|_2$ are defined to compare the registration performance of the algorithms. Here, $(\mathbf{A}^*, \mathbf{t}^*)$ represents the true affine transformation between model and data sets, and (\mathbf{A}, \mathbf{t}) denotes the affine transformation esti-

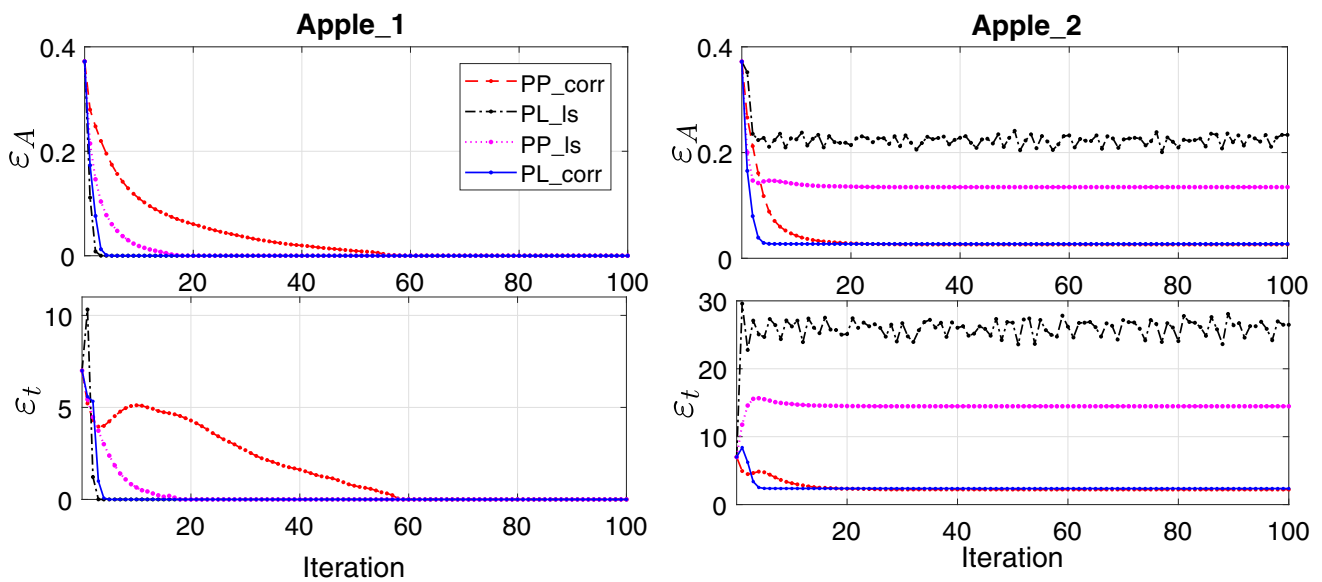


Fig. 7 For the affine registration algorithms, the number of iterations required to match the shapes

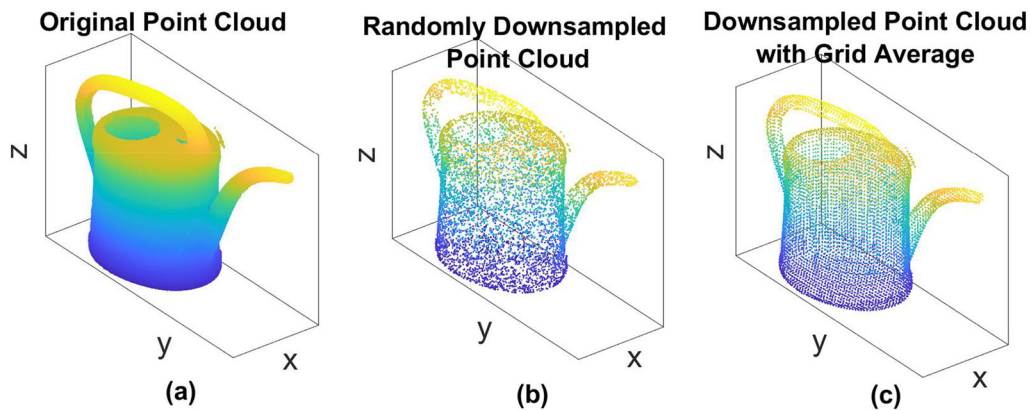


Fig. 8 Original point cloud and its down-sampled versions

Table 2 The number of points in the down-sampled 3D shapes, and the upper and lower boundaries of the 3D shapes registered in this study

Shape	Quantity	(z, \bar{z})	(y, \bar{y})	(x, \bar{x})
Dino	7841/7894	(−50, 50)	(−80, 90)	(−90, 50)
Dog	6175/6136	(−40, 50)	(−50, 50)	(−20, 30)
Lion	11961/11946	(−40, 40)	(−20, 20)	(−50, 80)
Watering Can	7350/7264	(−100, 110)	(−210, 130)	(−70, 70)

ated by the utilized algorithm. For modified model sets cases in 2D experiments, all compared numerical results are drawn in Fig. 14a for matrix \mathbf{A} and in Fig. 14b for translation vector \mathbf{t} . The results in the bar plots say that our algorithm is able to register the shapes in each case with reasonable error levels. This proves that our affine registration algorithm is also highly robust to outliers, noise effects, and/or major changes in the point sets.

The overall 2D and 3D affine registration performances of the algorithms are listed comparatively in Table 3. Here, it

is demonstrated that our algorithm (*AfJCP-PL-corr*) has the minimum registration errors in all cases in terms of having noise and/or missing data. Another important outcome of this table is that as the similarity ratio between the model set and the data set decreases, the estimation performance of the algorithms falls.

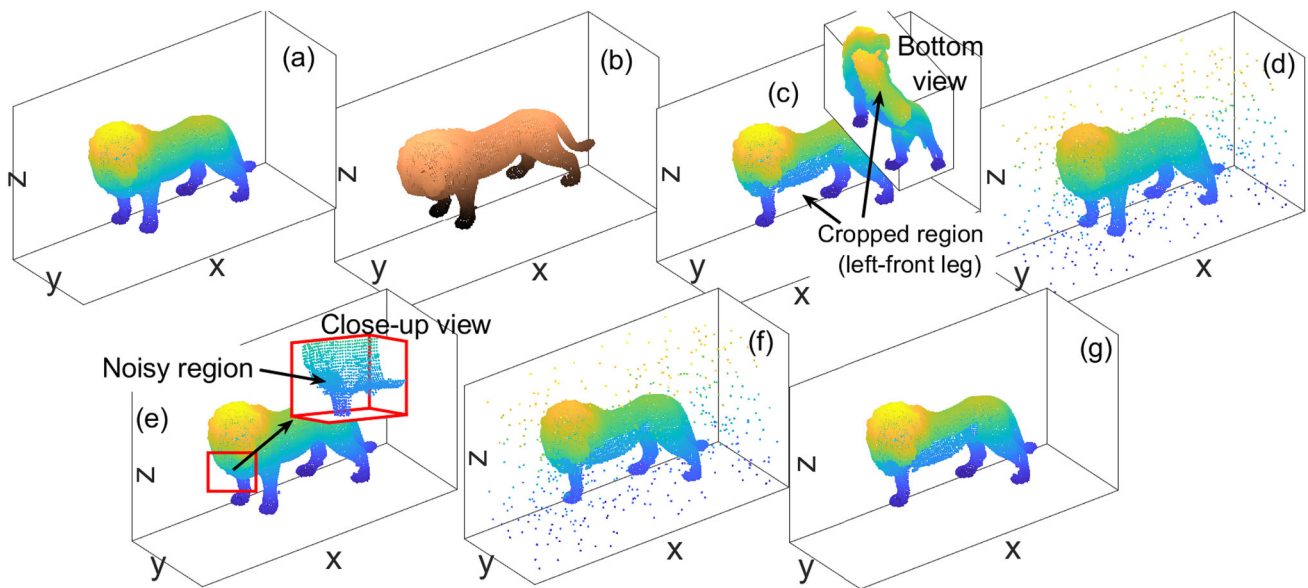


Fig. 9 Data and model sets in 3D before registration: lion example, (a) original MS, (b) data set, (c) cropped MS, (d) MS with uniform noise, (e) MS with Gaussian noise, (f) cropped MS with uniform noise, and (g) cropped MS with Gaussian noise

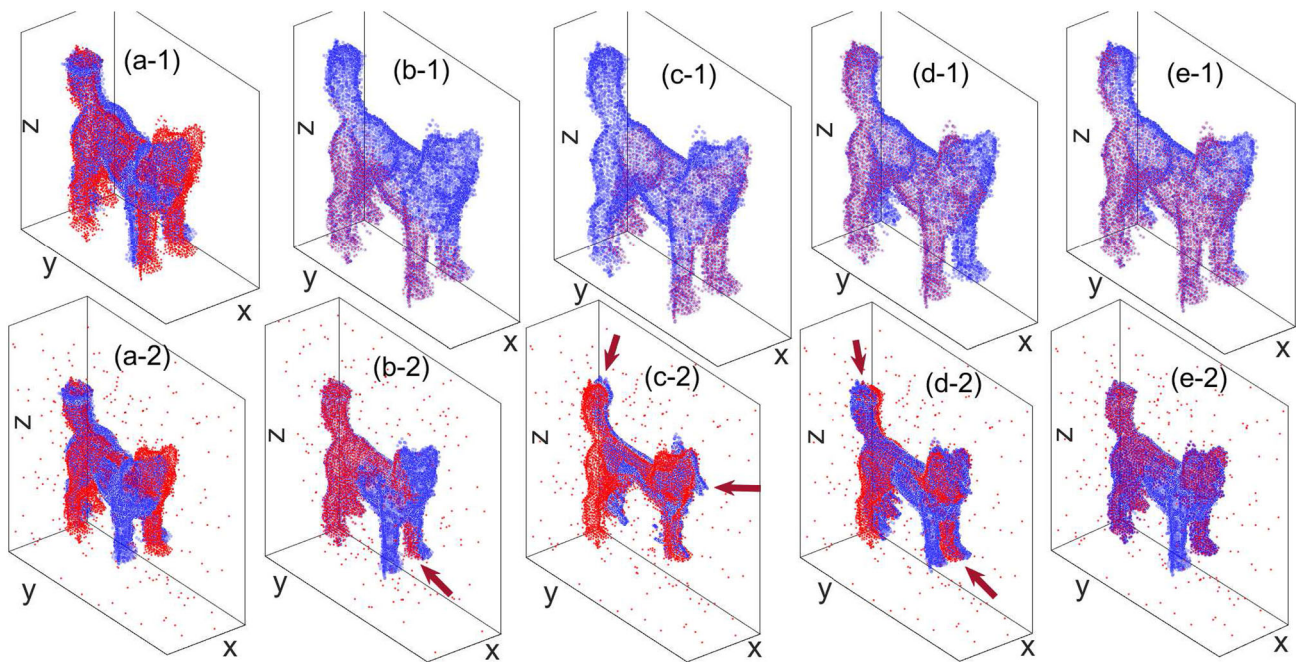


Fig. 10 Registration results of 3D dog shape: (a/e-1) original model set vs (a/e-2) model set having missing data and uniform noise - (a) the shapes before registration, registration results of (b) CPD, (c) AfJCP-PP-ls, (d) AfJCP-PL-ls and (e) AfJCP-PL-corr (our algorithm)

Success rates for 2D/3D registration experiments

Figure 15 shows the overall success rate, the number of required iterations on average, and the mean elapsed time per registration for affine registration of 2D and 3D shapes. Successful registration is defined as finding affine transformation with an error of less than 10%. On the other hand, the number of required iterations for a match is defined as

the iteration where transformation error falls below 1% otherwise equals the maximum number of iterations. The statistics presented in Figure 15 are generated by performing 576 separate experiments in 2D space and 108 separate experiments in 3D space. In affine registration, eight different 2D and four different 3D shapes having various model sets such as original, cropped, uniform or Gaussian noise inserted and/or combinations of them are utilized.

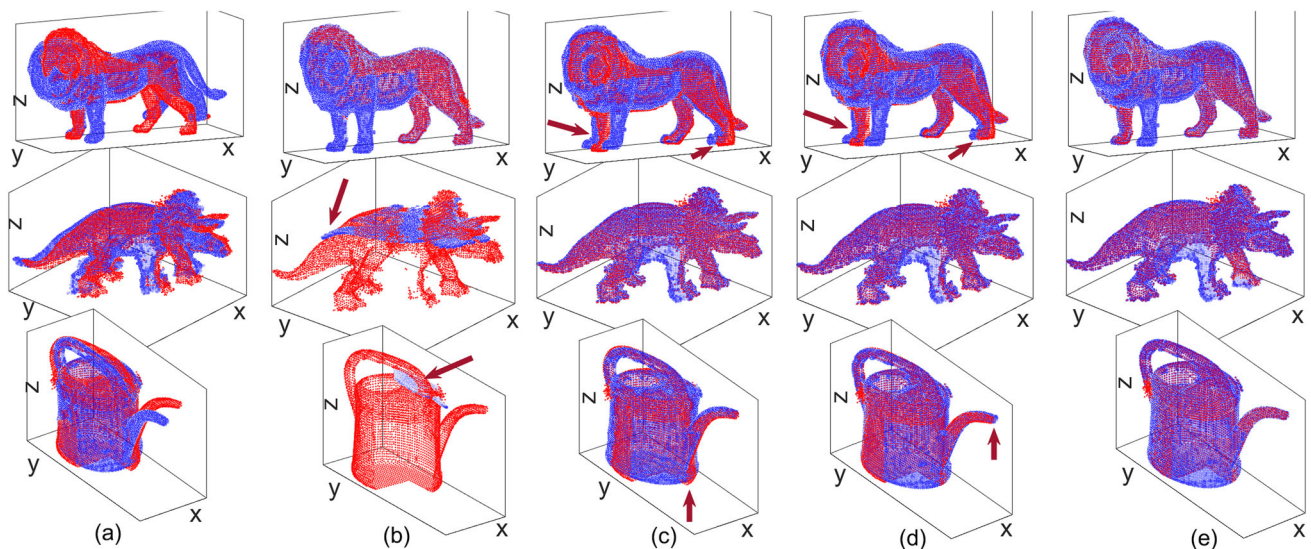


Fig. 11 Registration results of 3D shapes: model set having missing data and Gaussian noise (a) the shapes before registration, registration results of (b)CPD, (c) *AfICP-PP-ls*, (d) *AfICP-PL-ls* and (e) *AfICP-PL-corr* (our algorithm)

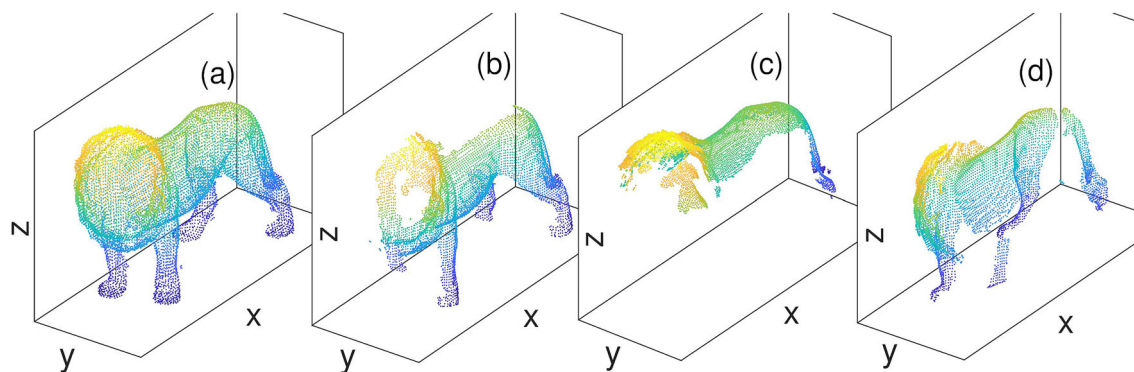


Fig. 12 The generated model sets based on 3D point cloud based on lion shape to see partial registration performance of our algorithm—(a) original model set, (b) Partial 1: front section of the lion, (c) Partial 2: top section of the lion, (d) Partial 3: rear section of the lion

It is deduced that our method is the most successful and highly robust algorithm in 2D and 3D affine registration among the algorithms compared in this study. In 2D registration, *AfICP-PP-corr* and our algorithm (*AfICP-PL-corr*), which use the correntropy criterion to construct objective function, have similar registration performance in terms of accuracy (the result of our algorithm is slightly better). However, in 3D registration, the estimation performance of *AfICP-PP-corr* decreases and it fails approximately 20% of the registration experiments (see in Fig. 15a).

AfICP-PL-ls, another algorithm that uses line/surface normals similar to *AfICP-PL-corr*, also requires fewer iterations to match the 2D/3D shapes than approaches using the point-to-point metric (see in Fig. 15b). Moreover, as seen in Fig. 15c, it is the fastest algorithm in 3D registration since the correntropy criterion brings extra computational load for *AfICP-PP-corr* and *AfICP-PL-corr*. However, rather than being fast, the main thing is the success of the algorithm in

matching and its capability to resist real-world factors such as noises/outliers. When Fig. 15a is examined, the results achieved demonstrate that the success rate of *AfICP-PL-ls* in 2D registration is terrible (around 40% successful matches), on the other hand, its success rate increases in 3D registration, but it cannot even reach success level of *AfICP-PP-corr*.

Our algorithm is also compared with the *CPD* algorithm, which is a robust and non-ICP-based registration method, for 3D point set experiments. It is apparent that *AfICP-PL-corr* is better in terms of not only success rate but also computational requirements. Since the *CPD* approach utilizes the GMM for representing the point sets, its registration performance is higher for randomly down-sampled point clouds than those down-sampled with grid average technique. However, our method developed in this study can tolerate both of them.

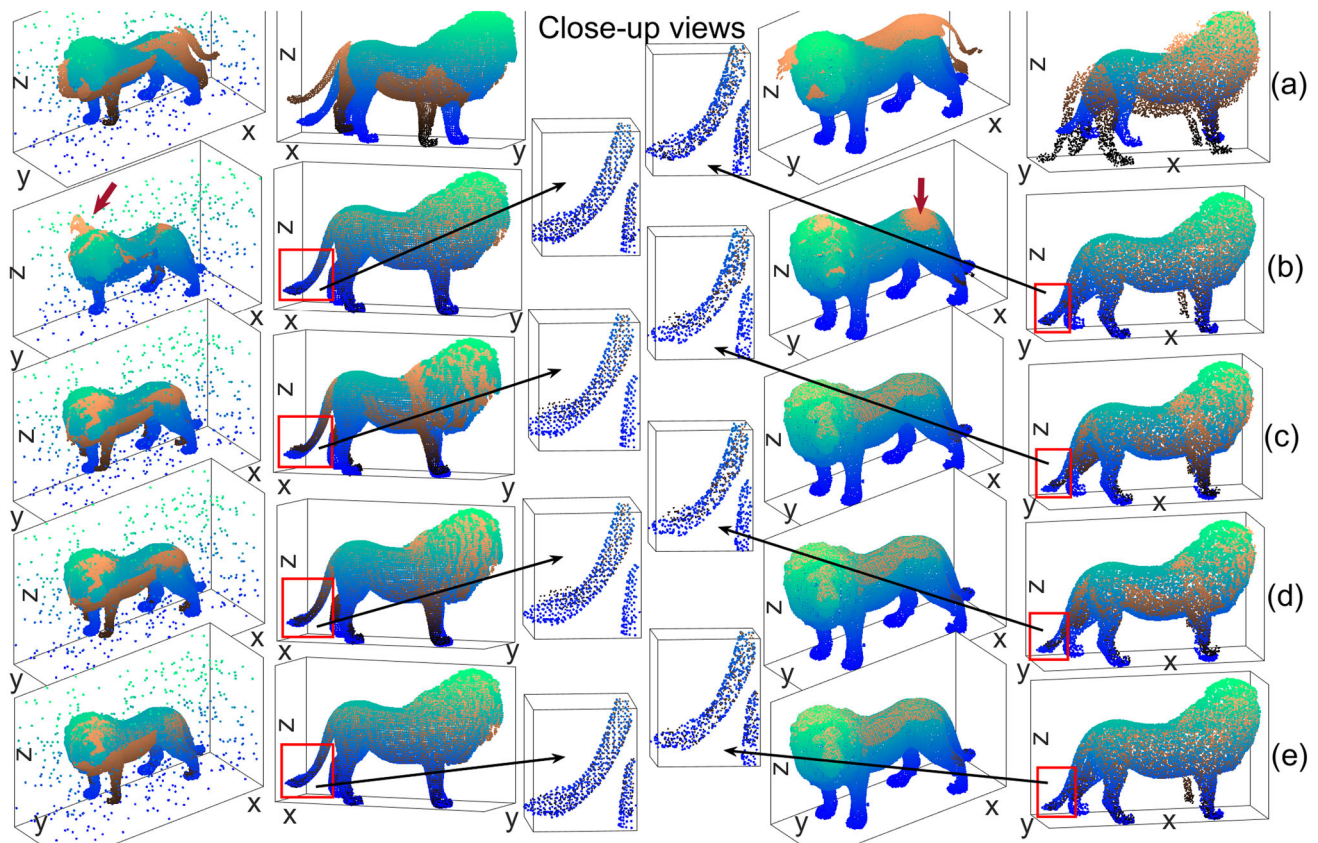
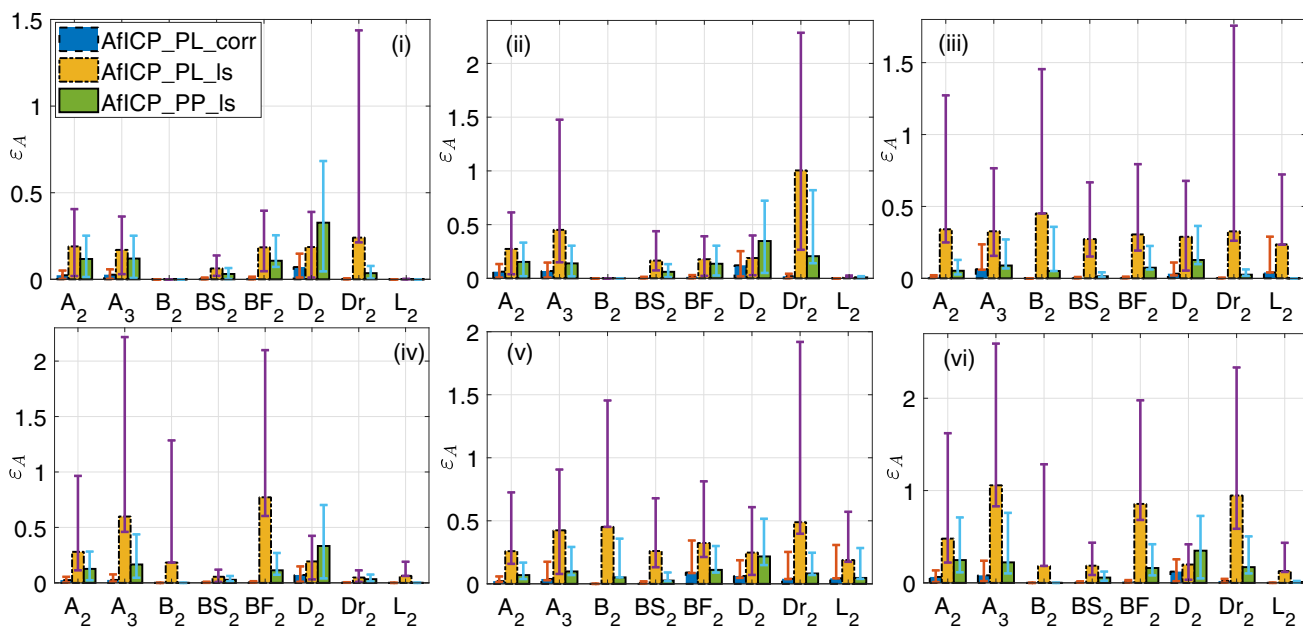


Fig. 13 Partial registration performance of the algorithms—3D model sets with different types: MS having uniform noise and missing data in column 1, MS having missing data in column 2, original partial MS in

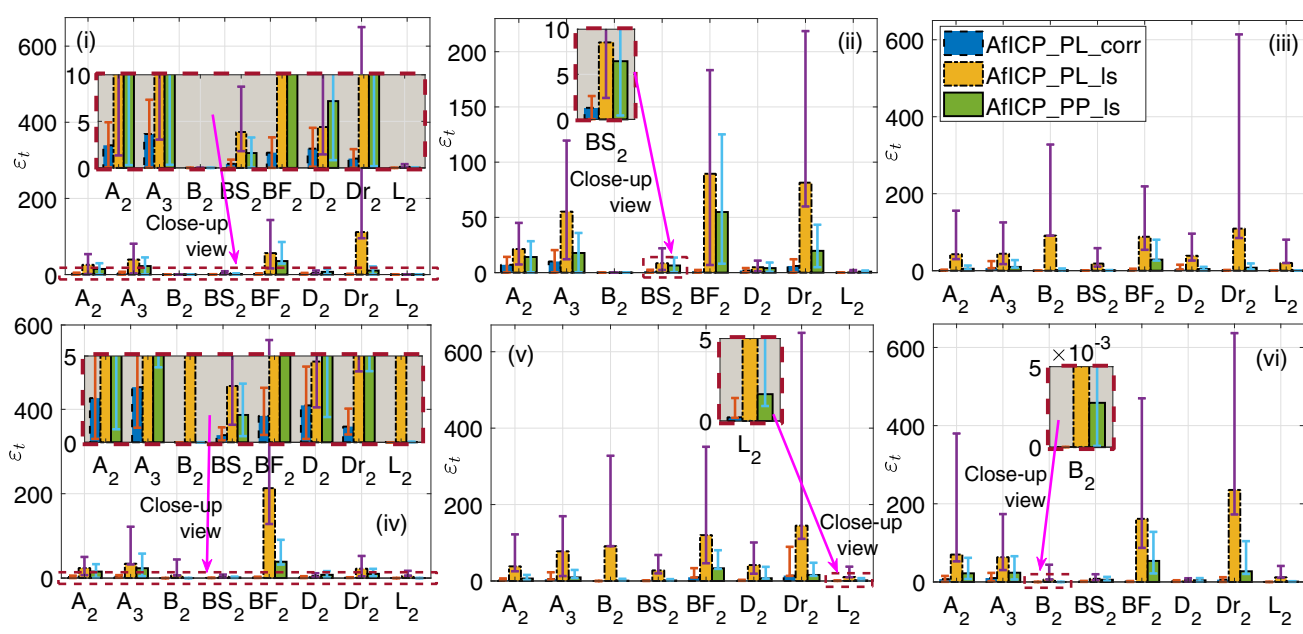
column 3, MS having missing data and down-sampled with random ds method; (a) before registration, (b) CPD, (c) AfICP-PP-ls, (d) AfICP-PL-ls, and (e) AfICP-PL-corr (our algorithm) performances

Table 3 The average 2D and 3D affine registration performances of the algorithms (CPD, AfICP-PP-ls, AfICP-PL-ls, and AfICP-PL-corr): the affine transformation estimation errors of the algorithms are shown in terms of (ϵ_A, ϵ_T)

Evaluating criteria and model set type		2D			
		AfICP_PP_ls	AfICP_PL_ls	AfICP_PL_corr (our algorithm)	
(ϵ_A, ϵ_T)	O	(0.046, 5.747)	(0.065, 15.097)	(0.008, 0.672)	
	C	(0.118, 13.285)	(0.295, 32.321)	(0.026, 2.244)	
	U	(0.031, 3.964)	(0.283, 57.165)	(0.010, 0.865)	
	G	(0.050, 6.057)	(0.172, 21.071)	(0.009, 0.701)	
	C,U	(0.072, 9.424)	(0.349, 72.005)	(0.029, 3.585)	
	C,G	(0.142, 14.270)	(0.418, 51.740)	(0.026, 2.143)	
Evaluating criteria and model set type		3D			
		CPD	AfICP_PP_ls	AfICP_PL_ls	AfICP_PL_corr (our algorithm)
(ϵ_A, ϵ_T)	O	(0.232, 2.095)	(0.010, 0.029)	$(\approx 1e-5, \approx 1e-4)$	$(\approx 1e-5, \approx 1e-4)$
	C	(0.309, 17.05)	(0.392, 4.119)	(0.231, 2.533)	(0.034, 0.392)
	U	(0.241, 2.193)	(0.011, 0.029)	$(\approx 1e-5, \approx 1e-4)$	$(\approx 1e-5, \approx 1e-4)$
	G	(0.398, 16.37)	(0.012, 6.031)	$(\approx 1e-5, \approx 1e-4)$	$(\approx 1e-5, \approx 1e-4)$
	C,U	(0.304, 16.88)	(0.309, 2.997)	(0.171, 1.888)	(0.021, 0.359)
	C,G	(0.436, 14.99)	(0.335, 3.446)	(0.216, 2.244)	(0.035, 0.390)



(a)



(b)

Fig. 14 **a** Matrix **A** and **b** translation vector **t** estimation error on affine registration of 2D point sets having major changes, A_2 : Apple 2, A_3 : Apple 3, B_2 : Bat 2, BS_2 : Bunny Stand 2, BF_2 : Butterfly 2, D_2 : Dog 2,

Dr_2 : Dragon 2, L_2 : Leaf 2 shapes—the results for (i) original MS, (ii) cropped MS, MS with (iii) uniform and (iv) Gaussian noise, cropped MS with (v) uniform and (vi) Gaussian noise

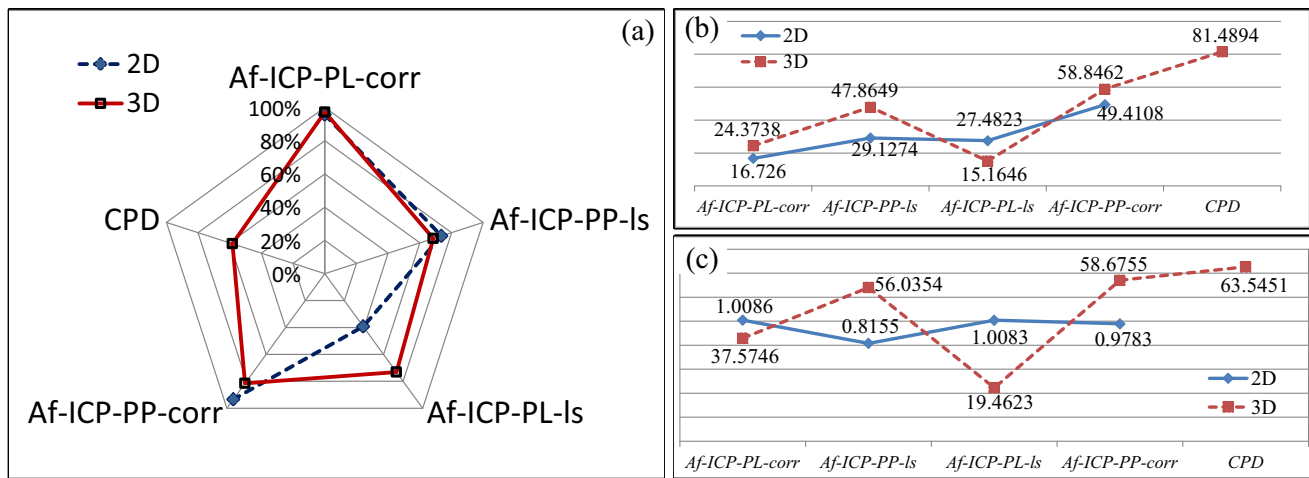


Fig. 15 For the algorithms used in 2D/3D point cloud registration, (a) the overall success rates of them, (b) the number of iterations required for each registration on average, and (c) mean elapsed time for each registration

Computational complexity analysis for AfICP-PL-corr

The computational load of the proposed approach is analyzed in Fig. 15 in terms of the average elapsed time and the number of iterations required per registration. However, the time needed to match point sets will vary depending on the properties of the hardware on which the algorithm is executed. Therefore, utilizing the $\mathcal{O}(\cdot)$ framework, the amount of work the CPU has to do for executing the *AfICP-PL-corr* algorithm will be explored based on the size of the input point set.

Let n be the number of points in the input point cloud, then the relationship between time elapsed in each iteration and n has been obtained as in Fig. 16a, b for 2D and 3D matching cases, respectively. The achieved linear curves in Fig. 16a, b shows that the computational complexity of the *AfICP-PL-corr* algorithm is $\mathcal{O}(n)$. On the other hand, it can be deduced from Fig. 16c that registration takes less time in 2D space than in 3D space even if the number of points in the sets is identical. There is a constant ratio between the matching times since the 2D and 3D registration curves almost overlap with each other when the time elapsed in 2D is scaled with the constant scalar $s_c = 4.57$. As $\mathcal{O}(s_c \times n)$ is equivalent to $\mathcal{O}(n)$ according to the properties of the $\mathcal{O}(\cdot)$ framework, the results confirm that the computational complexity of the developed algorithm is $\mathcal{O}(n)$.

Evaluation of AfICP-PL-corr registration performance for different outlier levels

The superior aspects of the affine ICP approach developed in this study, which uses the line/surface normals and correntropy criterion, have been demonstrated by extensive experiments on 2D and 3D data sets. In this sub-section, it is

aimed to investigate how the outlier elimination performance changes for different outlier levels. In this context, various degrees of outliers (10%, 20%, 30%, 50%, and 70%) with both uniform and Gaussian distributions were injected into 2D and 3D point clouds, and in this case, registration tests were performed. Some matching results for 2D and 3D point sets are given in Figs. 17 and 18, respectively. In total, 72 experiments were carried out for 2D and 96 experiments for 3D. It has been observed that the point sets were successfully matched in both two-dimensional and three-dimensional trials.

The achieved numerical results in Fig. 19 demonstrate that there is no significant relationship between the level of outliers and the registration error level of the method proposed in this study. However, the number of iterations required to match the point clouds, and thus the matching time, increase proportionally with the quantity of outliers.

As a result, the ICP variant proposed in this study, *AfICP-PL-corr*, is highly successful in affine registration of 2D and 3D point sets. Compared to other algorithms, although it is slow in terms of the time required for each iteration, the total registration duration is relatively shorter as it achieves the result (transformation) with less iteration and its success rate shows that *AfICP-PL-corr* is quite robust in point set registration.

Discussion

An ICP algorithm tries to estimate the transformation between two point sets by optimizing a cost function iteratively. However, except for special cases such as developed methods by [22,46], ICP algorithms converge to a locally optimal solution. Therefore, an initial guess is very important

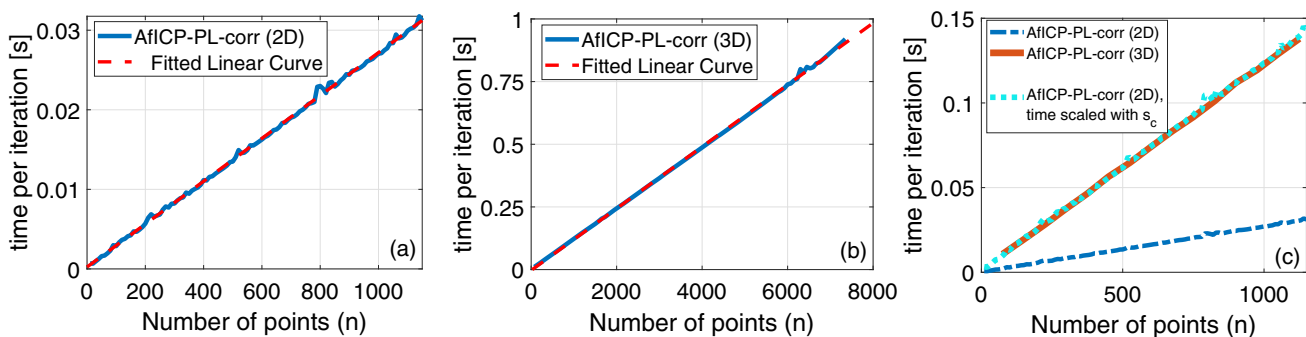


Fig. 16 The computational complexity of the developed *AfICP-PL-corr* algorithm: the average elapsed time per iteration vs the number of points in the point cloud for (a) 2D and (b) 3D registration, the computation time relationship between the 2D and 3D scenarios

for ICP algorithms to ensure that the correct transformation is going to be found. Before performing the point cloud registration process, it makes sense to overcome getting stuck in the local minimum problem by running an initial guess computation algorithm like covariance matrix-based approach by [47] or adding some constraints to the cost function such as bounding rotation angle (see [48,49] studies). However,

since the aim of this study is to show the matching performance of the proposed registration algorithm under effects such as noises, deformations, and/or outliers, an initial guess algorithm was not used in experimental verification tests. By keeping the amount of rotation between the two-point sets registered at low levels, the divergence of the algorithms originating from the initial guess error has been prevented.

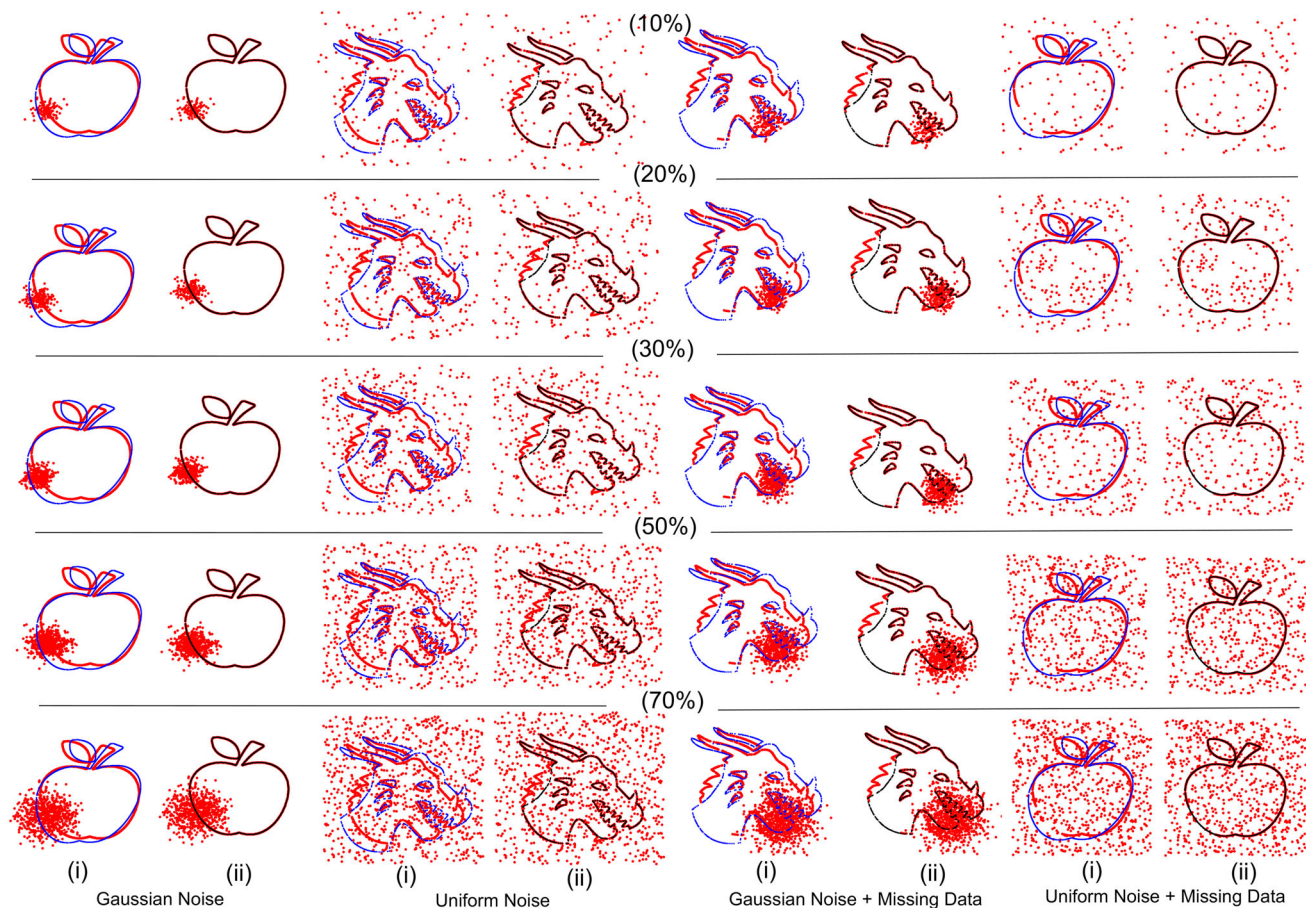


Fig. 17 Registration results of 2D shapes for varying outlier levels: the number of outliers is 10%, 20%, 30%, 50%, and 70% of the number of points in the original model set—(i) the shapes before registration and (ii) registration results of *AfICP-PL-corr* (our algorithm)

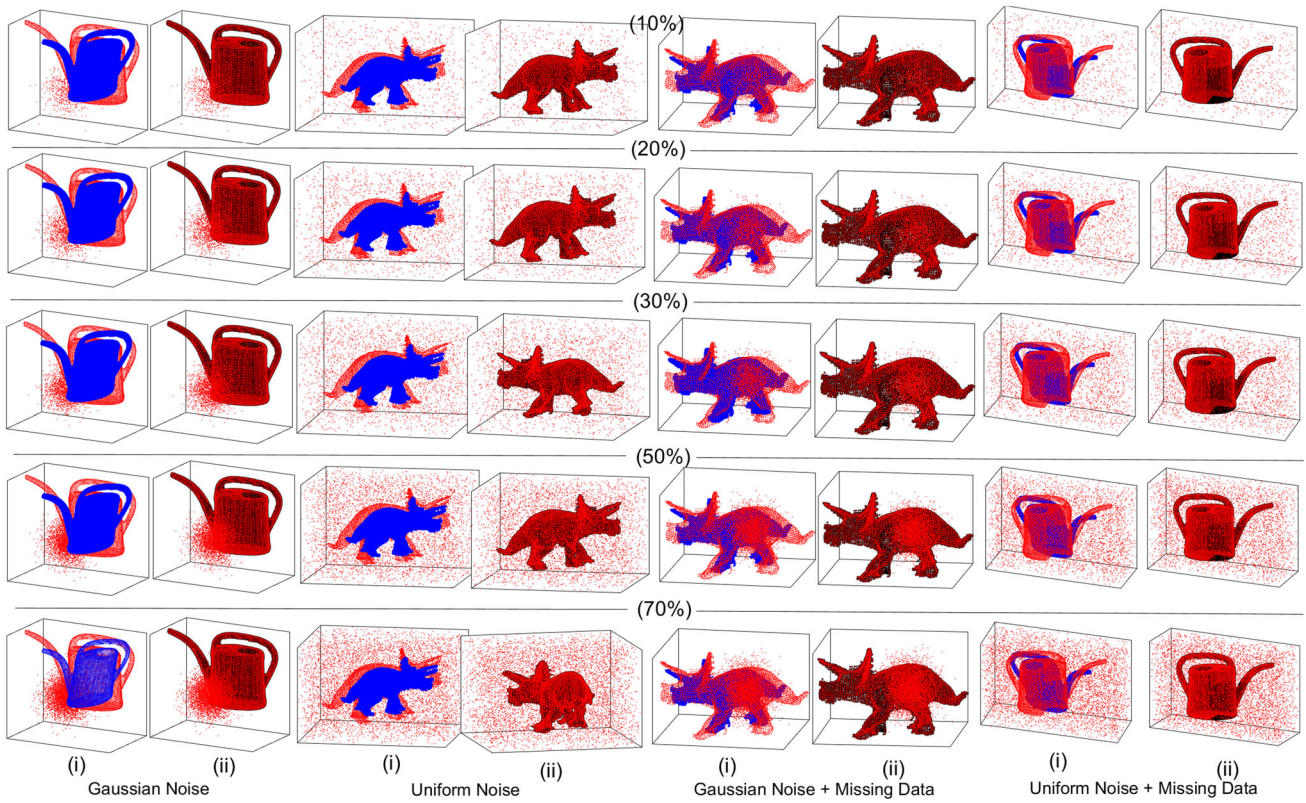


Fig. 18 Registration results of 3D shapes for varying outlier levels: the number of outliers is 10%, 20%, 30%, 50% and 70% of the number of points in the original model set—(i) the shapes before registration and (ii) registration results of *AfICP-PL-corr* (our algorithm)

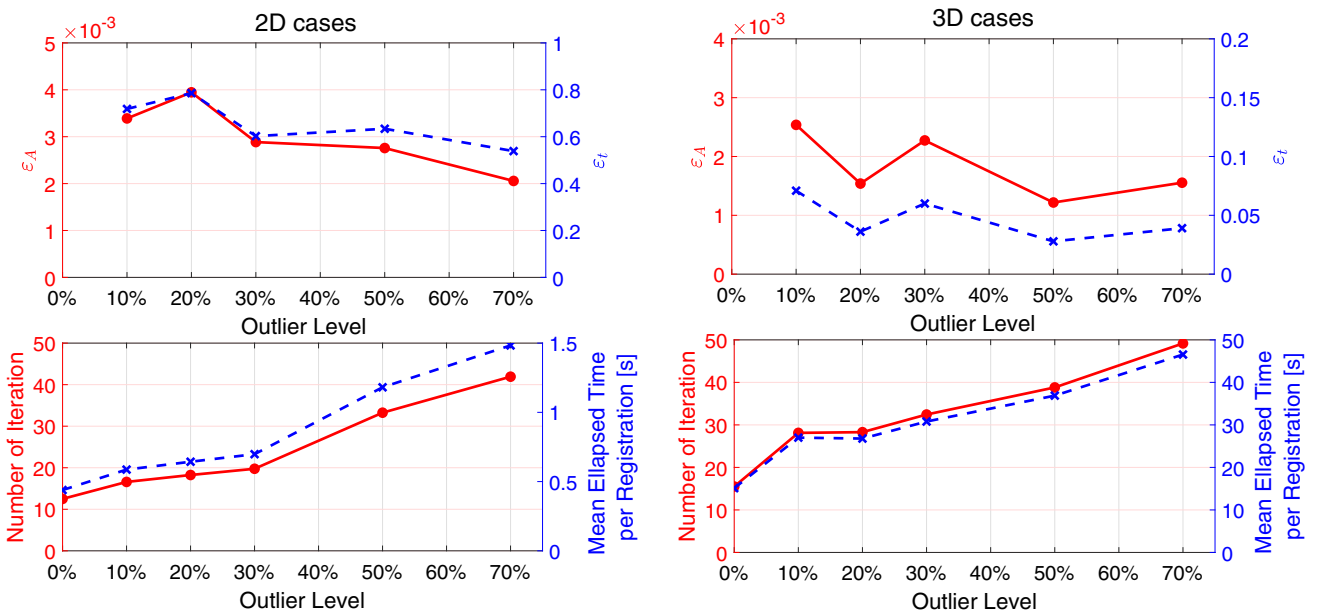


Fig. 19 Numerical comparison for varying outlier levels: 2D registration (top-left) and 3D registration (top-right) error rates, variation of the average number of iterations and time required for each 2D (bottom-left) and 3D (bottom-right) registration attempt according to different outlier levels

In all experimental studies, the identity matrix was selected as the initial transformation.

For the algorithms utilized for state-of-the-art comparisons and our algorithm developed in this study, the average elapsed time for each registration is given for 2D and 3D registration cases in Fig. 15c. When these results shown in the time domain are analyzed, it is seen that the matches take quite a long time. These periods are much longer, especially for 3D registrations where the number of points in the point clouds is high. However, it should be noted that the speeds and success rates of the algorithms were evaluated with respect to each other only. As a result, no optimization has been made for the algorithms or no attention has not been paid to coding to make the algorithms faster. Coding was done using basic functions available in the MATLAB environment such as matrix multiplication and inverse operations. In addition, the PC where the tests are performed is insufficient in terms of computing capacity. Currently, it may be thought that this algorithm is not suitable for real-time applications, but it is possible to take some precautions to run the algorithms faster for real-time works.

Conclusions

In this study, a robust affine ICP variant using the correntropy criterion and line/surface normals is presented to solve the affine point cloud registration problem. Correntropy-based affine ICP variants are highly robust to noises, outliers, and missing data effects, and with our implementation of line/surface normals in the construction of objective function, the number of required iterations to find the transformation between point clouds has been reduced and registration performance in terms for accuracy has been increased. Optimization of the objective function defined to compute current transformation is given in detail as a closed-form solution. Our method (*AfICP-PL-corr*), some other affine ICP variants (*AfICP-PP-ls*, *AfICP-PL-ls*, *AfICP-PP-corr*), and *CPD* are employed in 2D and 3D point cloud registration experiments for comparison, 594 separate experiments on 8 different shapes in 2D and 132 separate experiments on 4 different shapes in 3D, and it is achieved that our algorithm is better in terms of robustness and accuracy rate. As shown, our algorithm also works well in partial registration problems.

The study, which is given in [12], applied affine ICP for the indoor precise localization problem of mobile robots. In future studies, we would like to deploy this affine ICP variant developed in this study for the localization problem. Moreover, in the current version of our method, the transformation between point clouds is found by locally optimizing the objective function. However, finding a globally optimal solution set is highly important not to diverge. For that reason, any modification and/or enhancement can be deployed on our

method such as bidirectional distance extension approach for ICP algorithms or global optimization approaches. Similarly, the initial guess is crucial for the ICP algorithm to make proper estimations. This pre-process can be attached to the method proposed in this study since it is essential especially for successful registration of the point clouds having a high amount of rotation difference between each other.

Acknowledgements This study was fully supported by the Scientific and Technological Research Council of Turkey, TUBITAK, under grant number 116E734.

Declaration

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Gao Y, Wang M, Tao D, Ji R, Dai Q (2012) 3-D object retrieval and recognition with hypergraph analysis. *IEEE Trans Image Process* 21(9):4290. <https://doi.org/10.1109/TIP.2012.2199502>
2. Takimoto RY, Tsuzuki MdSG, Vogelaar R, de Castro Martins T, Sato AK, Iwao Y, Gotoh T, Kagei S (2016) 3D reconstruction and multiple point cloud registration using a low precision RGB-D sensor. *Mechatronics* 35:11. <https://doi.org/10.1016/j.mechatronics.2015.10.014>
3. Lan S, Guo Z, You J (2019) A non-rigid registration method with application to distorted fingerprint matching. *Pattern Recognit* 95:48. <https://doi.org/10.1016/j.patcog.2019.05.021>
4. Yeo D, Lee CO (2020) Variational shape prior segmentation with an initial curve based on image registration technique. *Image Vis Comput* 94. <https://doi.org/10.1016/j.imavis.2019.103865>
5. Krüger J, Schultz S, Handels H, Ehrhardt J (2020) Registration with probabilistic correspondences—Accurate and robust registration for pathological and inhomogeneous medical data. *Comput Vis Image Understanding* 190. <https://doi.org/10.1016/j.cviu.2019.102839>
6. Jing W, Polden J, Tao PY, Lin W, Shimada K (2016) View planning for 3d shape reconstruction of buildings with unmanned aerial vehicles, In 2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV) (IEEE, 2016), pp. 1–6. <https://doi.org/10.1109/ICARCV.2016.7838774>
7. Li C, Lu B, Zhang Y, Liu H, Qu Y (2018) 3D reconstruction of indoor scenes via image registration. *Neural Process Lett* 48(3):1281. <https://doi.org/10.1007/s11063-018-9781-0>

8. Kajihara T, Funatomi T, Makishima H, Aoto T, Kubo H, Yamada S, Mukaigawa Y (2019) Non-rigid registration of serial section images by blending transforms for 3D reconstruction. *Pattern Recognit* 96. <https://doi.org/10.1016/j.patcog.2019.07.001>
9. Eppenhof KA, Lafarge MW, Moeskops P, Veta M, Pluim JP (2018) Deformable image registration using convolutional neural networks, In *Medical Imaging 2018: Image Processing*, vol. 10574 (International Society for Optics and Photonics, 2018), vol. 10574, p. 105740S. <https://doi.org/10.1117/12.2292443>
10. Ma Y, Guo Y, Lei Y, Lu M, Zhang J (2017) Efficient rotation estimation for 3D registration and global localization in structured point clouds. *Image Vision Comput* 67:52. <https://doi.org/10.1016/j.imavis.2017.09.003>
11. Servos J, Waslander SL (2017) Multi-Channel Generalized-ICP: A robust framework for multi-channel scan registration. *Robot Autonomous Syst* 87:247. <https://doi.org/10.1016/j.robot.2016.10.016>
12. Yilmaz A, Temeltas H (2021) Integration of affine ICP into the precise localization problem of smart-AGVs: Procedures, enhancements and challenges. *Trans Institute Measurement Control* 43(8):1695. <https://doi.org/10.1177/0142331220933430>
13. Huang X, Mei G, Zhang J, Abbas R (2021) A comprehensive survey on point cloud registration, A comprehensive survey on point cloud registration, arXiv preprint [arXiv:2103.02690](https://arxiv.org/abs/2103.02690)
14. Choy C, Dong W, Koltun V (2020) in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2514–2523. <https://doi.org/10.1109/CVPR42600.2020.00259>
15. Bai X, Luo Z, Zhou L, Chen H, Li L, Hu Z, Fu H, Tai CL (2021) PointDSC: Robust Point Cloud Registration using Deep Spatial Consistency, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15,859–15,869
16. Huang S, Gojic Z, Usvyatsov M, Wieser A, Schindler K (2021) PREDATOR: Registration of 3D Point Clouds with Low Overlap, In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4267–4276
17. Wan T, Du S, Cui W, Yao R, Ge Y, Li C, Gao Y, Zheng N (2021) RGB-D point cloud registration based on salient object detection. *IEEE Trans neural Netw Learn Syst*. <https://doi.org/10.1109/TNNLS.2021.3053274>
18. Li M, Xu RY, Xin J, Zhang K, Jing J (2020) Fast non-rigid points registration with cluster correspondences projection. *Signal Process* 170. <https://doi.org/10.1016/j.sigpro.2019.107425>
19. Besl PJ, McKay ND et al (1992) A method for registration of 3-D shapes. *IEEE Trans Pattern Anal Mach Intell* 14(2):239. <https://doi.org/10.1109/34.121791>
20. Censi A, An ICP variant using a point-to-line metric, In *2008 IEEE International Conference on Robotics and Automation (IEEE, 2008)*, pp. 19–25. <https://doi.org/10.1109/ROBOT.2008.4543181>
21. Segal A, Haehnel D, Thrun S (2009) Generalized-ICP, in *Proceedings of Robotics: Science and Systems (Seattle, USA)*. <https://doi.org/10.15607/RSS.2009.V.021>
22. Yang J, Li H, Campbell D, Jia Y (2015) Go-ICP: a globally optimal solution to 3D ICP point-set registration. *IEEE Trans Pattern Anal Mach Intell* 38(11):2241. <https://doi.org/10.1109/TPAMI.2015.2513405>
23. Bellekens B, Spruyt V, Berkvens R, Weyn M (2014) A survey of rigid 3d pointcloud registration algorithms, In *Fourth International Conference on Ambient Computing, Applications, Services and Technologies, Proceedings (IARA, 2014)*, pp. 8–13
24. Zinßer T, Schmidt J, Niemann H (2005) Point set registration with integrated scale estimation, In *International conference on pattern recognition and image processing*, pp. 116–119
25. Du S, Zheng N, Xiong L, Ying S, Xue J (2010) Scaling iterative closest point algorithm for registration of m-D point sets. *J Vis Commun Image Represent* 21(5–6):442. <https://doi.org/10.1016/j.jvcir.2010.02.005>
26. Du S, Zheng N, Ying S, Liu J (2010) Affine iterative closest point algorithm for point set registration. *Pattern Recognit Lett* 31(9):791. <https://doi.org/10.1016/j.patrec.2010.01.020>
27. Makovetski A, Voronin S, Kober V, Tihonkih D (2017) Affine registration of point clouds based on point-to-plane approach. *Procedia engineering* 201:322. <https://doi.org/10.1016/j.proeng.2017.09.635>
28. Horaud R, Forbes F, Yguel M, Dewaele G, Zhang J (2010) Rigid and articulated point registration with expectation conditional maximization. *IEEE Trans Pattern Anal Mach Intell* 33(3):587. <https://doi.org/10.1109/TPAMI.2010.94>
29. Chen H, Zhang X, Du S, Wu Z, Zheng N (2019) A correntropy-based affine iterative closest point algorithm for robust point set registration. *IEEE/CAA J Automatica Sinica* 6(4):981. <https://doi.org/10.1109/JAS.2019.1911579>
30. Du S, Xu G, Zhang S, Zhang X, Gao Y, Chen B (2020) Robust rigid registration algorithm based on pointwise correspondence and correntropy. *Pattern Recognit Lett* 132:91. <https://doi.org/10.1016/j.patrec.2018.06.028>
31. Zhang X, Jian L, Xu M (2018) Robust 3D point cloud registration based on bidirectional Maximum Correntropy Criterion. *PLoS One* 13(5). <https://doi.org/10.1371/journal.pone.0197542>
32. Du S, Liu J, Zhang C, Zhu J, Li K (2015) Probability iterative closest point algorithm for mD point set registration with noise. *Neurocomputing* 157:187. <https://doi.org/10.1016/j.neucom.2015.01.019>
33. Chen H, Wu Z, Du S, Zhou N, Sun J (2016) Robust scale iterative closest point algorithm based on correntropy for point set registration, In *2016 Australian Control Conference (AuCC) (IEEE, 2016)*, pp. 238–242. <https://doi.org/10.1109/AUCC.2016.7868195>
34. Du S, Zhang C, Wu Z, Liu J, Xue J (2016) Robust isotropic scaling ICP algorithm with bidirectional distance and bounded rotation angle. *Neurocomputing* 215:160. <https://doi.org/10.1016/j.neucom.2015.05.144>
35. Wu Z, Chen H, Du S, Fu M, Zhou N, Zheng N (2019) Correntropy based scale ICP algorithm for robust point set registration. *Pattern Recognit* 93:14. <https://doi.org/10.1016/j.patcog.2019.03.013>
36. Wang G, Wang Z, Chen Y, Zhao W (2015) A robust non-rigid point set registration method based on asymmetric gaussian representation. *Comput Vis Image Understand* 141:67. <https://doi.org/10.1016/j.cviu.2015.05.014>
37. Wang G, Zhou Q, Chen Y (2017) Robust non-rigid point set registration using spatially constrained Gaussian fields. *IEEE Trans Image Process* 26(4):1759. <https://doi.org/10.1109/TIP.2017.2658947>
38. Min Z, Wang J, Meng MQH (2020) Robust generalized point cloud registration with orientational data based on expectation maximization. *IEEE Trans Auto Sci Eng* 17(1):207. <https://doi.org/10.1109/TASE.2019.2914306>
39. Myronenko A, Song X (2010) Point set registration: coherent point drift. *IEEE Trans Pattern Anal Mach Intell* 32(12):2262
40. Min Z, Meng MQH (2020) Robust and Accurate Nonrigid Point Set Registration Algorithm to Accommodate Anisotropic Positional Localization Error Based on Coherent Point Drift. *IEEE Trans Auto Sci Eng*. <https://doi.org/10.1109/TASE.2020.3027073>
41. Hill DL, Batchelor PG, Holden M, Hawkes DJ (2001) Medical image registration. *Phys Med Biol* 46(3):R1. <https://doi.org/10.1088/0031-9155/46/3/201>
42. Wu Z, Chen H, Du S (2016) Robust affine iterative closest point algorithm based on correntropy for 2D point set registration, In *2016 International Joint Conference on Neural Networks (IJCNN) (IEEE)*, pp. 1415–1419. <https://doi.org/10.1109/IJCNN.2016.7727364>
43. Barber CB, Dobkin DP, Huhdanpaa H (1996) The quickhull algorithm for convex hulls. *ACM Trans Math Softw (TOMS)* 22(4):469. <https://doi.org/10.1145/235815.235821>

44. Greenspan M, Yurick M (2003) Approximate kd tree search for efficient ICP, In Fourth International Conference on 3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings. (IEEE), pp. 442–448. <https://doi.org/10.1109/IM.2003.1240280>
45. Glira P, Pfeifer N, Briese C, Ressel C (2005) A Correspondence Framework for ALS Strip Adjustments based on Variants of the ICP Algorithm Korrespondenzen für die ALS-Streifenausgleichung auf Basis von ICP, Photogrammetrie-Fernerkundung-Geoinformation **2015**(4), 275. <https://doi.org/10.1127/pfg/2015/0270>
46. Lei H, Jiang G, Quan L (2017) Fast descriptors and correspondence propagation for robust global point cloud registration. IEEE Trans Image Proces 26(8):3614. <https://doi.org/10.1109/TIP.2017.2700727>
47. Feng Z (2019) An efficient initial guess for the ICP method. Pattern Recogni Lett 125:721. <https://doi.org/10.1016/j.patrec.2019.07.019>
48. Zhang C, Du S, Liu J, Xue J (2016) Robust 3D point set registration using iterative closest point algorithm with bounded rotation angle. Signal Process 120:777. <https://doi.org/10.1016/j.sigpro.2015.01.021>
49. Liu X, Zhu L, Liu X, Lu Y, Wang X (2017) Hierarchical skull registration method with a bounded rotation angle, In International Conference on Intelligent Computing (Springer), pp. 563–573. https://doi.org/10.1007/978-3-319-63315-2_49

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