A Hierarchical Bayesian Model for Matching Unlabeled Point Sets

Xin Hu[1], Deqiong Ding[2], Dianhui Chu[1], Xiaodong Zhang[1], Xuequan Zhou[1], Hua Zhang[1], and Chunshan Li[1]
{*hithuxin@hit.edu.cn*[1], *mathddq@hit.edu.cn*[2], *chudh@hit.edu.cn*[3]}

School of Computer Science and Technology, Harbin Institute of Technology at Weihai, Weihai, China[1], Department of Mathematics, Harbin Institute of Technology at Weihai, Weihai, China[2].

Abstract. Point set registration is the key in many scientific disciplines. Target at several challenges in registration (e.g. initial registration, outliers, missing data, and local trap), we propose a robust registration method for two point sets using a hierarchical Bayesian model, which is combined with Markov chain Monte Carlo inference. Our approach is based on the introduction of a template of hidden locations underlying the observed configuration points. A Poisson process prior is assigned to these locations, resulting in a simplified formulation of the model. We make use of a structure containing the relevant information on the matches. We conduct several experiments to demonstrate that our algorithm is accurate and robust.

Keywords: Hierarchical model · Markov chain Monte Carlo · Matching · Registration.

1 Introduction

In many scientific disciplines, one is confronted with the problem of comparing objects for object recognition, especially in the scenario of medical image analysis [1–3]. The aim of object recognition is to correctly identify objects in a scene and estimate their pose (location and orientation). The object recognition usually involves comparing point sets (configurations), where the points in different configurations are often unlabeled in the sense that there is no natural correspondence between the points in each configuration. Numerous techniques [4–9] have been studied over years for the geometrical comparison of objects, where the configurations are labeled, or partly labeled, that is, when the point correspondences between the objects under study have been established. Here, we concern about the configurations unlabeled, as the the correspondences between the points of each configuration are unknown. Therefore, the problem becomes pair matching: identifying and labeling corresponding points in different configurations, so called point set registration.

The Iterative Closest Point (ICP) algorithm, proposed by Besl and McKay [10], and Chen and Medioni [11], is an accurate and efficient approach which is first proposed to solve this problem. The ICP algorithm defines an energy function, which can indicate

MOBIMEDIA 2019, June 29-30, Weihai, People's Republic of China

DOI 10.4108/eai.29-6-2019.2282677

the summation of squared distances for point pairs in two sets. Then by minimizing this energy function, ICP finds the optimal transformation. However, ICP could only solve rigid registration problem since it requires a rough initial registration, which is applicable to only rigid registration, and is sensitive to outliers and missing data.

To overcome these challenges, many researchers have extended the original ICP algorithm to deal with non-rigid registration. One approach is to provide initial transformations close to the optimal solution, such as principal component analysis and independent component analysis [12–14]. However, these methods could not be extended to solve the affine registration problem. Addressing this challenge, Zha *et al.* [15] extended the rigid registration problem to isotropic scale deformation. Du *et al.* [16, 17] also proposed an affine registration method based on the ICP algorithm; Ying *et al.* [12, 12] proposed the so-called Scale-ICP algorithm targeting at scale problem. Liu [18, 19] added geometric constraints and weights to the ICP algorithm. Although many efforts have also been made to improve the accuracy for the point pairs, all the algorithms mentioned above no longer work when two point sets exist a large number of outliers or missing data.

Target at outliers and missing data issue, Chetverikov *et al.* [20] proposed the Trimmed ICP (TrICP) algorithm, which can tackle the low overlapping problem partially. The TrICP algorithm picks out outliers and then conducts ICP by minimizing the Trimmed Squared Distance (TSD). Based on the TSD, Phillips *et al.* [21] extended the ICP to a fast automatic overlap rate estimation method, which can can identify and discard outliers. Du *et al.* [22, 23] tried to solve the low overlapping problem by introducing an overlapping percentage and a scale factor into a least-square function, which is based on bidirectional distance in the isotropic scaling registration. However, these algorithms could not deal with affine registration problem. They may fail due to the anisotropic scale transformation.

Addressing the anisotropic scale transformation problem, recently several improved algorithms of ICP by adopting Lie groups have been reported in the literature [16, 29]. These algorithms have the advantage over the anisotropic scale transformation problem partially by converging monotonically to a local minimum. Although these modified ICP algorithms improve the convergence issue of local minima and sensitivity to outliers and disturbances, they still suffer from slow convergence or even divergence, if proper initial pose estimate is not available [25].

Addressing the challenges of unlabeled configurations, outliers, and missing data, this paper presents a hierarchical Bayesian model from getting alignment trapped in local modes, and proposes a Markov chain Monte Carlo (MCMC) algorithm for making inference on this model. We organize the paper as follows. In Section 2, we introduce the hierarchical Bayesian model. Section 3 describes the inference of MCMC algorithm on the model. Experiments and analysis are shown in Section 4. And finally Section 5 concludes the paper.

2 Hierarchical Bayesian Model

2.1 Hierarchical Model with Poisson Process Assumption

We consider two configurations, M and N, whose points are recorded in d-dimensional space. These two point sets can be wrote as $M = \{x_i,\ i = 1, 2, ..., m\}$ and $N = \{y_j,\ j = 1, 2, ..., n\}$, where m and n are the numbers of points in the configurations M and N, respectively. The labeling here is assumed to be arbitrary, which provides no initial information on the correspondences between points of M and N. Addressing the outliers and missing data challenges, we construct a template, $\mu = \{\mu_i,\ i = 1, 2, ..., w\} \in R^d$, to delineate the correspondence between the two configurations, where w is unknown. This template μ can be considered as a hidden point set including the missing data of the two configurations, which can be interpreted as the true locations of the two configurations. On the other hand, the configuration points with outliers in M and N can be considered as noisy observations of the template μ.

In order to specify this template clearly, we define two sets of labeling arrays, $\xi = \{\xi_i,\ i = 1, 2, ..., w\}$ and $\eta = \{\eta_j,\ j = 1, 2, ..., w\}$, which link the indexes of the configuration points to that of their corresponding template points, where ξ_i and η_j are the indexes of the template μ underlying the configuration points. Then we can associate the two configurations by associating the two index sets ξ_i and η_j within the same template. For one configuration, we assume that a template point can be observed at most once, which means the rest points of the template μ remain unobserved.

Here, we assume that each configuration goes through some transformations before being observed. Let $A = \{A_1, A_2\}$ be the transformations, which bring the points of the two configurations back to the template points. Under this assumption, our alignment model can be written as

$$\begin{aligned} A_1 \cdot x_i &= \mu_{\xi_i} + \varepsilon_i \\ A_2 \cdot y_j &= \mu_{\eta_j} + \varepsilon_j \end{aligned} \tag{1}$$

where x_i and y_j are the points of configurations M and N, respectively. ε_i and ε_j are the random errors, which are assumed to have density f and to be independent of the template points and of all the other errors.

Since μ is the template interpret the true locations of the configurations, there can be many options for μ, including the following multiple estimates:

(a) μ may include the two configurations M and N.
(b) μ may include one of the two configurations, say M.
(c) μ may include one configuration N.
(d) μ may include none of the two configurations.

Suppose that the number of template points in case (a) is L, then we can estimate the configuration M has $m - L$ points in case (b), the configuration N has $n - L$ points in case (c), and the template μ has $w - m - n + L$ points remain in case (d). Furthermore, we note the probabilities of points included in these cases as p_a, p_b, p_c, and p_d, respectively. The estimation of μ is a random distribution, which subjects to Poisson distribution. The correspondence of these three probabilities can be expressed as $p_a = \gamma \cdot p_b \cdot p_c$, where γ is the previous information about the configurations M and N. Based on the Poisson process assumption on the template μ, the count L is an independent Poisson variable

with mean $\gamma\lambda v p_b p_c$, where λ and v are the Poisson parameters. The prior distribution for the estimation counts can be expressed as

$$p(L|m,n) = \frac{e^{-\gamma\lambda v p_b p_c}(\lambda v \gamma p_b p_c)^L}{L!} \times \frac{e^{-\lambda v p_b}(\lambda v p_b)^{m-L}}{(m-L)!} \times \frac{e^{-\lambda v p_c}(\lambda v p_c)^{n-L}}{(n-L)!} \tag{2}$$

where λ and v are Poisson parameters, $\lambda v p_b$ and $\lambda v p_c$ are the mean counts for cases (b) and (c).

From Equation 2, we can rewrite the estimation probabilities for cases (b) and (c) as

$$\begin{aligned} p(m) &= p_b^m \\ p(n) &= p_c^n \end{aligned} \tag{3}$$

Using Equation 2 and 3, the form of the prior distribution for the estimation counts (Equation 2) can be rewrote as

$$p(L) \propto \frac{(\gamma/\lambda v)^L}{(m-L)!(n-L)!L!} \tag{4}$$

where symbol "$\propto$" indicates proportionality with respect to the variables to the left of the conditioning sign.

In order to specify the correspondence between the configurations(M and N) and the template μ, we use a match matrix Λ, which is a $m \times n$ matrix of 1s and 0s. This match matrix Λ is defined as

$$\Lambda_{i,j} = \begin{cases} 1, & \xi_i = \eta_j \\ 0, & others \end{cases} \tag{5}$$

where ξ_i and η_j are the indexes of the template μ underlying the observations (configuration points). With this match matrix, we can specify the correspondence between the two configurations M and N. If $\Lambda_{i,j} = 1$, then template μ has the true location of the two configurations, as described in case (a). Otherwise $\Lambda_{i,j} = 0$, then template μ does not have a point to interpret the locations for the two configurations, which is described in cases (b), (c), and (d). Based on the definition of match matrix Λ, every row or column of this matrix sums to 1 or 0. And we can have $\sum_{i,j} \Lambda_{i,j} = L$. Based on Equation 4, we can have the likelihood of this match matrix as following

$$\begin{aligned} p(\Lambda) = p(\lambda|L)p(L) &= \frac{p(L)}{L!\binom{m}{L}\binom{n}{L}} \\ &\propto \frac{(\gamma/\lambda v)^L}{(m-L)!(n-L)!L!} \cdot \frac{1}{L!\binom{m}{L}\binom{n}{L}} \\ &\propto (\gamma/\lambda v)^L \end{aligned} \tag{6}$$

2.2 Joint Bayesian Model

After constructing template μ and defining match matrix Λ, now we seek for the joint likelihood of μ and the transformation $A = \{A_1, A_2\}$. We start from the estimate of

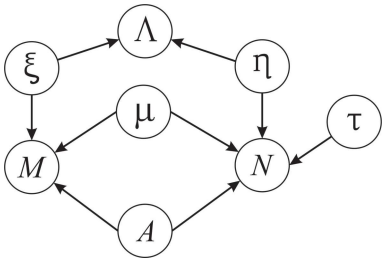


Fig. 1. Directed acyclic graph of the hierarchical model.

the template μ. For the points of configuration M in case (b), we find their distribution has the following form

$$p(x|\Lambda, A) = \prod_{i:\Lambda_{i,j}=0,\forall j} v^{-1} \int_{R^d} |A_1| h_1(x_i - \mu) d\mu \tag{7}$$

where $\{i : \Lambda_{i,j} = 0, \forall j\}$ is the description of case (b), x is the point belongs to configuration M subjecting to this condition, A_1 is the transformation matrix, $|A_1|$ denotes the absolute value of the determinant of the matrix A_1, h_1 is the error density of ε_i.

The same to case (c), the distribution of the points included in case (c) has the form

$$p(y|\Lambda, A) = \prod_{j:\Lambda_{i,j}=0,\forall i} v^{-1} \int_{R^d} |A_2| h_2(y_j - \mu) d\mu \tag{8}$$

where $\{j : \Lambda_{i,j} = 0, \forall i\}$ is the description of case (c), y is the point belongs to configuration N subjecting to this condition, $|A_2|$ denotes the absolute value of the determinant of the matrix A_2, h_2 is the error density of ε_j.

After discovering the distribution forms for cases (b) and (c), we can have the distribution for case (a) as following

$$\begin{aligned} &p(\mu|\Lambda, A) = \\ &\textstyle\prod_{i,j:\Lambda_{i,j}=1} v^{-1} \int_{R^d} |A_1| h_1(x_i - \mu) |A_2| h_2(y_j - \mu + \tau) d\mu \end{aligned} \tag{9}$$

where $\{i, j : \Lambda_{i,j} = 1\}$ is the description of case (a), τ is the parameter of shift transformation.

From Equation 7, 8, and 9, the posterior distribution has the form

$$\begin{aligned} &p(x, y|\Lambda, A) \propto \\ &v^{L-(m+n)} |A_1|^m |A_2|^n \textstyle\prod_{i,j:\Lambda_{i,j}=1} \alpha(A_1 x_i - A_2 y_j - \tau) \end{aligned} \tag{10}$$

where α is the error density of $\varepsilon_i - \varepsilon_j$.

So far, we have the hierarchical model, including the construction of the template μ. And we display this model in the directed acyclic graph (Fig. 1).

Combine Equation 6 and 10, we finally get the joint posterior distribution corresponding with the match matrix Λ,

$$\begin{aligned} &p(\Lambda, M, N|A) \propto \\ &|A_1|^m |A_2|^n \prod_{i,j:\Lambda_{i,j}=1} \alpha\gamma(A_1 x_i - A_2 y_j - \tau)/\lambda \end{aligned} \tag{11}$$

There is an assumption about the error density in Poisson process, proposed by Green and Mardia *et. al* [26], that the error densities are centered Gaussian densities with covariance matrices all equal to σ^2, where σ is the standard deviation. Based on this assumption, we consider α, h_1 and h_2 subject to Gaussian distribution. From Equation 1 and 11, we find the following form for our joint posterior model with the Gaussian assumption for the errors,

$$\begin{aligned} &p(\Lambda, A, M, N) \propto \\ &|A_1|^m |A_2|^n p(A)p(\tau)p(\sigma) \prod_{i,j:\Lambda_{i,j}=1} \frac{\gamma \times \Phi((A_1 x_i - A_2 y_j - \tau)/\sigma\sqrt{2})}{\lambda(\sigma\sqrt{2})^2} \end{aligned} \tag{12}$$

where "Φ" denotes Gaussian function with standard deviation σ, which is the error variance.

3 Markov Chain Monte Carlo Inference

Here we want to make inference on the parameters of the joint Bayesian model we get (Equation 12). In this equation, the parameters we concern are the match matrix Λ, the translations A, and the error variance σ. This model is a joint distribution, whose unwieldy aspect makes it difficult to progress in inference, using conventional analytic or numerical estimation methods. Thus, we simulate a Markov chain by using MCMC [27] simulation, which updates the parameters in sweeps. In the sweeps, the underlying transition kernel of the Markov chain verifies detailed balance. Based on this simulation, our model shown in Equation 12 can be considered to be stationary, constraint, and distribution.

3.1 Updating the Translation A

For the transformation parameter A, conditionally conjugate priors can be found, which result in full conditional distributions of the same form. This character can make the updating of these parameters straightforward relatively. We can assume the translations A are rotation matrices. Then the conjugate priors can be found for the rotation matrices: $p(A) \propto exp(tr(F^T \cdot A))$, where F is some $d \times d$ matrix, $tr(\cdot)$ the trace operator. Based on this conjugate assumption, we have $|A| = 1$. And the involved calculation yields the conditional distribution of rotation matrices $A = \{A_1, A_2\}$ as following

$$\begin{aligned} &p(A|\Lambda, M, N, \sigma, \tau) \\ &\propto |A_1|^m |A_2|^n p(A) \prod_{i,j:\Lambda_{i,j}=1} \Phi((A_1 x_i - A_2 y_j - \tau)/\sigma\sqrt{2}) \\ &\propto p(A) exp\left(\sum_{i,j:\Lambda_{i,j}=1} -\frac{\|A_1 x_i - A_2 y_j - \tau\|^2}{4\sigma^2}\right) \\ &\propto p(A) exp\left(\frac{\sum_{i,j:\Lambda_{i,j}=1}(A_1 x_i - \tau)A_2 y_j}{2\sigma^2}\right) \\ &\propto p(A) exp\left\{tr\left[\frac{\sum_{i,j:\Lambda_{i,j}=1}(A_1 x_i - \tau)A_2 y_j}{2\sigma^2}\right]\right\} \end{aligned} \tag{13}$$

Rather than updating the rotation matrices themselves, we will work on the corresponding rotation angles. We concentrate on the $d = 3$ dimensional case here. For $d = 3$ case, we have three generalized Euler angles θ_{12}, θ_{23}, and θ_{13}. We assume that the rotation matrices are with conjugate matrix Fisher prior. Since the relative rotation $A_1^T \cdot A_2$ is uniform and mutually independent, it is reasonable for us to make such assumption. Based on this assumption, we can simplify the rotation matrices as $A = A_1^T \cdot A_2$. Then the model (Equation 13) can be rewrote as

$$p(A|\Lambda, M, N, \sigma, \tau) \propto p(A) exp\left\{tr\left[\frac{\sum_{i,j:\Lambda_{i,j}=1}(Ax_i - \tau)y_j}{2\sigma^2}\right]\right\} \tag{14}$$

where A can be represented as $A = R_z(\theta_{12})R_y(\theta_{13})R_x(\theta_{23})$, R_x, R_y, and R_z are rotation matrices. These rotation matrices have the descriptions as following

$$\begin{aligned} R_x(\theta_{23}) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\theta_{23} & sin\theta_{23} \\ 0 & -sin\theta_{23} & cos\theta_{23} \end{pmatrix} \\ R_y(\theta_{13}) &= \begin{pmatrix} cos\theta_{13} & 0 & sin\theta_{13} \\ 0 & 1 & 0 \\ -sin\theta_{13} & 0 & cos\theta_{13} \end{pmatrix} \\ R_z(\theta_{12}) &= \begin{pmatrix} cos\theta_{12} & sin\theta_{12} & 0 \\ -sin\theta_{12} & cos\theta_{12} & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \tag{15}$$

where $\theta_{12} \in [-\pi, \pi)$, $\theta_{13} \in [-\pi/2, \pi/2]$, and $\theta_{23} \in [-\pi, \pi)$. We take a Metropolis-Hastings jump to update these Euler angles. The acceptance probability of a jump can be calculated from Equation 14 as following

$$p_{jump} = \min\{1,\ exp\left\{tr\left[\frac{\sum_{i,j:\Lambda_{i,j}=1}(Ax_i - \tau)y_j}{2\sigma^2}\right]\right\}\theta_{13}\} \tag{16}$$

3.2 Updating Continuous Parameters

Now we are going to update the continuous parameters τ (the shift transformation) and σ (the standard deviation of Gaussian density). As we mentioned, we assign an inverse gamma prior distribution to σ^2: $\sigma^{-2} \sim \Gamma(a, b)$, where $\Gamma(a, b)$ is the inverse gamma prior distribution, a and b are the shape and rate parameters of the gamma distribution, respectively. From Equation 12, 13, and 14, the full conditional distribution of σ^{-2} is

$$\begin{aligned} &p(\sigma^{-2}|\Lambda, A, \tau, M, N) \sim \\ &\Gamma(a + L/4, b + (1/4)\textstyle\sum_{i,j:\Lambda_{i,j}=1} \|x_i - Ay_j - \tau\|^2) \end{aligned} \tag{17}$$

With this full conditional distribution, the error variance can be updated by using a Gibbs sampler step, which can be simulated from the full conditional inverse gamma distribution.

Since the Gaussian priors are assigned to the translation matrices, we also suppose a Gaussian priori to the parameter of shift transformation τ as $\tau \sim N_d(\mu_N, \sigma_N^2)$, where

μ_N and σ_N are the mean and the standard deviation of the Gaussian distribution. Then we can obtain the full conditional distribution of τ as

$$p(\tau|\Lambda, A, \sigma, M, N) \sim N_d\left(\frac{\mu_N/\sigma_N^2+\sum_{i,j:\Lambda_{i,j}=1}(x_i-Ay_j)/2\sigma^2}{1/\sigma_N^2+L/2\sigma^2}, \frac{1}{1/\sigma_N^2+L/2\sigma^2}\right) \tag{18}$$

3.3 Updating the Match Matrix

The match matrix will be updated with a Metropolis-Hastings jump. We initiate configurations M and N to be in case (d) for matching. To update the match matrix Λ, we move the points in case (d) into case (a), (b), and (c). A point in case (d) can be selected randomly, the match will be assigned to a point in case (d) with the largest probability, then this pair of points can be moved to case (a). If a point is in case (a), the probability of removing it into case (b) or (c) is p_{reject}. Then the probability of matching it to another point in case (d) is $(1 - p_{reject})/(n^* - 2)$, where n^* is the number of points in N remain in case (d). We accept the new jump proposal (updated match matrix), Λ^*, with probability

$$p_{\Lambda^*} = \min\left\{1, \frac{\phi(\Lambda^*|\tau, \sigma, M, N) \cdot q}{\phi(\Lambda|\tau, \sigma, M, N) \cdot q^*}\right\} \tag{19}$$

where q and q^* are the parameters corresponding to Λ and Λ^*, respectively. They have the following relationship,

$$q/q^* \approx \begin{cases} p_{reject} \cdot n^*, & if\ making\ an\ unmatched\ point\ matched; \\ 1/(p_{reject} \cdot n^*), & if\ making\ a\ matched\ point\ unmatched; \\ 1, & if\ making\ a\ matched\ point\ match\ to\ a\ different\ point. \end{cases} \tag{20}$$

By using Equation 20, we can focus the update process on p_{reject} as following,

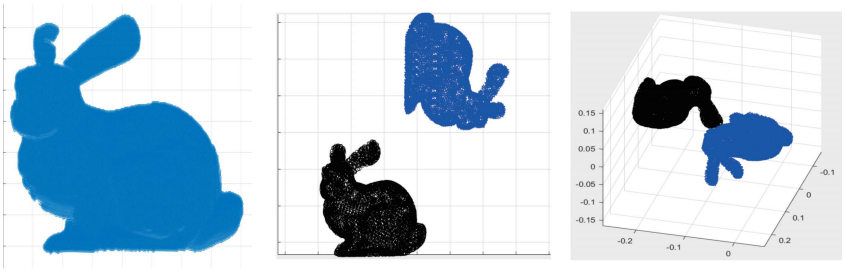
$$p_{reject} = \frac{\gamma \times \Phi((x_i - Ay_j - \tau)/\sigma\sqrt{2})}{\lambda(\sigma\sqrt{2})^2} \tag{21}$$

4 Experiments

In order to show the accuracy of our algorithm, we compare it with the Iterative Closest Point (ICP) algorithm, which is a representative method for point set registration, by using the Stanford Repository [1], Parasaurolophus and T-Rex meshes datasets from the UWA 3-D Modeling Dataset [28] and Skeleton Hand and Dragon from the Large Geometric Models Archive at Georgia Tech [2]. During our experiments, parameters were set as follows: $\alpha = 8$, $\gamma = 0.7$, $\lambda = 6$, $\sigma = 10^{-6}$, $\mu_N = 0.5$, and $\sigma_N = 10^{-6}$. The proposed algorithms were implemented in Matlab 2015a, using Intel(R) CPU i7-8550U @1.8 GHz with 8 GB RAM. All experiments were performed on the same computer.

[1] The stanford 3D scanning repository. http://graphics.stanford.edu/data/3Dscanrep/.

[2] The Large Geometric Models Archive. http://www.cc.gatech.edu/projects/large_models.



(a) (b) (c)

Fig. 2. The simulated point set of Stanford Bunny dataset. (a) The original shape of Stanford Bunny, (b) The simulated point sets in 2-D, (c) The simulated point sets in 3-D.

4.1 Datasets

The point sets were generated to simulate large number of outliers and missing data, by deleting 40% points in the original dataset randomly and adding 20% outliers. Two point sets for each registration were generated by the following procedures: (1) Copying two point sets from the original dataset, deleting 40% points of them randomly, adding 20% points to the point sets; (2) Rotating and shifting one copy randomly, the shift transformation range falls into [500, 20000] voxels. Therefore, we have two different point sets for each experiment. For each dataset in the three databases, we generated 50 pairs of simulated point sets. Here we take one pair of them generated from the Stanford Bunny for example (Fig. 2).

The iteration of match matrix update was set to 1000. Also take one pair of point sets generated from Stanford Bunny for example, the computation time of our method is around 3700 seconds, while that of ICP is around 5300 seconds. And the rotation matrix A and the parameter of shift transformation τ we obtained are shown as following

$$A = \begin{pmatrix} 0.94344 & -0.11864 & 0.30960 \\ -0.12571 & 0.73605 & -0.66515 \\ -0.30679 & -0.66644 & 0.67951 \end{pmatrix}$$

$$\tau = (0.08035, 0.03880, 0.06607)$$

We also obtained the rotation matrix R and the parameter of shift transformation T for ICP,

$$R = \begin{pmatrix} -0.14373 & -0.98920 & 0.02878 \\ -0.98913 & 0.14269 & -0.03549 \\ 0.030100 & -0.03356 & -0.99900 \end{pmatrix}$$

$$T = (0.14501, 0.13656, 0.09361)$$

4.2 Evaluation Metrics and Statistical Tests

In order to evaluate the proposed methods, registration error and overlap rate for each pair of simulated point sets are calculated. The registration error is borrowed from

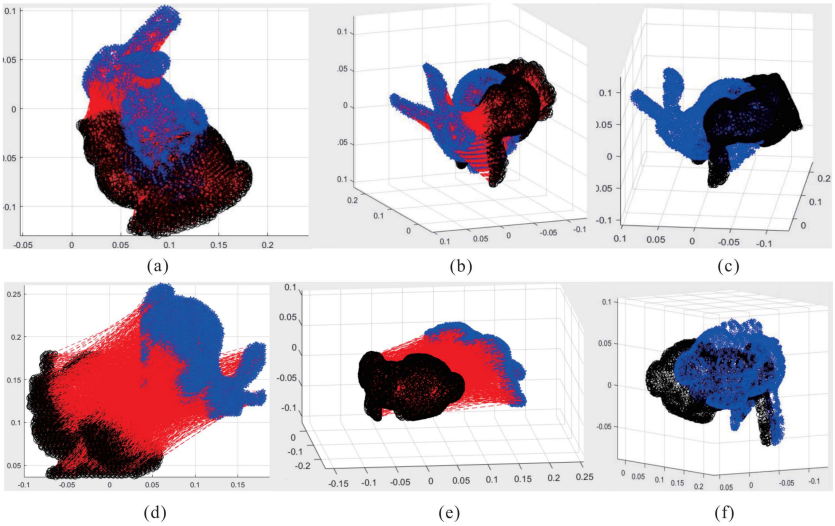


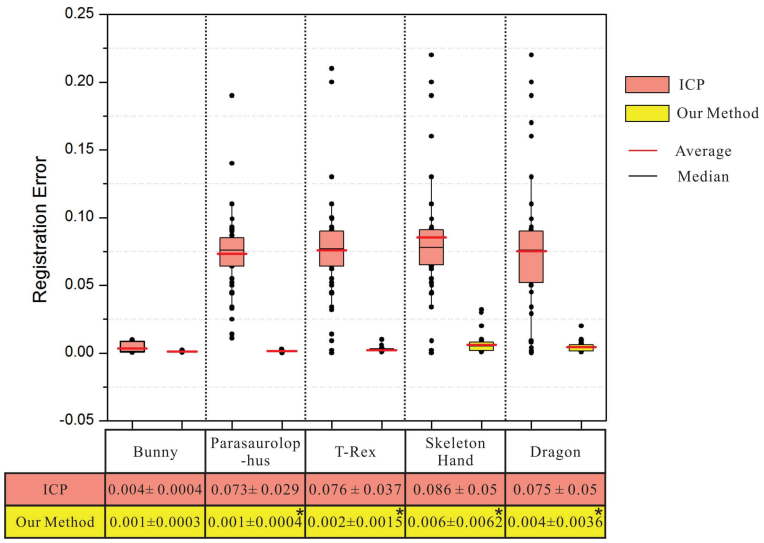
Fig. 3. The comparison results for Bunny. (a) The matching result of ICP in 2-D, (b) The matching result of ICP in 3-D, (c) The registration result of ICP in 3-D, (d) The matching result of our method in 2-D, (e) The matching result of our method in 3-D, and (f) The registration result of our method in 3-D.

Dong's paper [29], which can represent the matching rate and the transformation error. And we use a fast automatic overlap rate estimation technique [22] to calculate the overlap rate, which can represent the transformation error between the two simulated point sets. For registration error (overlap rate), the smaller (larger) the value is, the better the performance result.

In all the comparison experiments between our algorithm and ICP, the paired T-test is conducted to assess whether the difference in registration accuracy between the two proposed algorithms was statistically significant, with a significance level set at $p < 0.05$.

5 Result

The comparison results for the Stanford Bunny are shown in Fig. 3, including the matching results and registration results. The quantitative analysis of registration error and overlap rate are provided in Fig. 4 and Fig. 5, respectively. Observe from Fig. 3 and Fig. 4, our method performs much better than ICP, an average registration error of 0.004 ± 0.0004, 0.073 ± 0.029, 0.076 ± 0.037, 0.086 ± 0.05, and 0.075 ± 0.05 is obtained by ICP, while these values are changed to 0.001 ± 0.0003, 0.001 ± 0.0004, 0.002 ± 0.0015, 0.006 ± 0.0062, and 0.004 ± 0.0036 by our method. Fig. 3 shows that the matching rate of ICP is high in some cases, however, the registration error is still high due to the large number of outliers and missing data. In the opposite, our method

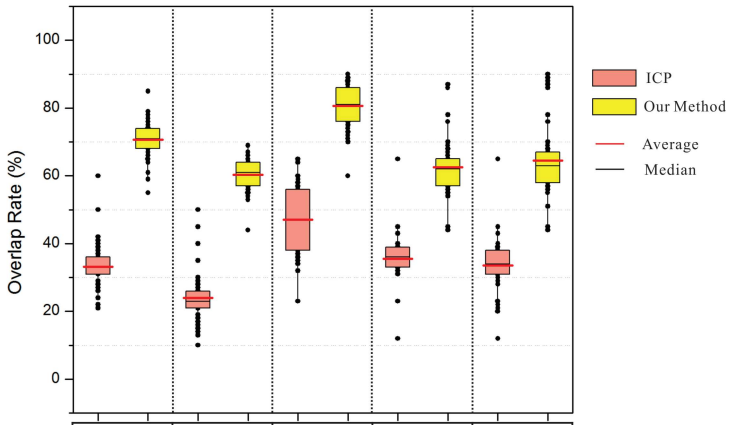


	Bunny	Parasaurolop -hus	T-Rex	Skeleton Hand	Dragon
ICP	0.004± 0.0004	0.073± 0.029	0.076 ± 0.037	0.086 ± 0.05	0.075 ± 0.05
Our Method	0.001±0.0003	0.001±0.0004*	0.002±0.0015*	0.006±0.0062*	0.004±0.0036*

Fig. 4. Summary of registration evaluation on the 5 datasets for ICP and our methods. For each dataset, we generate 50 pairs of simulated point sets. Registration errors are calculated from each pair of point set. In the table, $*$ indicates the statistically significant differences between ICP and our method at a significance level of 0.05.

achieves satisfactory results in most cases. The results shown in Fig. 4 demonstrate that our method is more resistant to outliers and missing data comparing to ICP, and the performance differences between our method and ICP are statistically significant ($p < 0.05$). We find that our method obtain reasonable overlap result, but ICP fail to do so. Fig. 5 also shows the capability of our method in facing outliers and missing data in another way. The overlap rates of our method are more reasonable than those of ICP. On the 5 datasets, the average overlap rates of ICP are 33.22 ± 6.92%, 23.94 ± 8.00%, 47.04 ± 10.77%, 35.56 ± 75.2%, and 33.54 ± 8.49%. The average overlap rates of our method are much higher, 72.16 ± 5.20%, 60.34 ± 4.61%, 80.60 ± 6.45%, 62.60 ± 8.20%, and 64.56 ± 10.27%.

In our experiments, we find that the outliers and missing data do affect the performance of our method. We show the performance of our method in the worst cases for a snapshot, Fig. 6. In some cases, the registration error of our method is high, not to mention the overlap rate, especially for the Skeleton Hand dataset. For the Dragon dataset, We found that the missing data can lead to large transformation error, even the matching rate is high, as shown in Fig. 6 Skeleton Hand and Dragon.



	Bunny	Parasaurolop -hus	T-Rex	Skeleton Hand	Dragon
ICP	33.22 ± 6.92	23.94 ± 8.00	47.04 ± 10.77	35.56 ± 75.2	33.54 ± 8.49
Our Method	72.16 ± 5.20*	60.34 ± 4.61*	80.60 ± 6.45	62.60 ± 8.20*	64. 56±10.27*

Fig. 5. Summary of overlap evaluation on the 5 datasets for ICP and our methods. Overlap rates are also calculated from each pair of point set. In the table, $*$ indicates the statistically significant differences between ICP and our method at a significance level of 0.05.

6 Conclusions and future work

We have proposed a hierarchical Bayesian model for objects alignment, and the corresponding inference by using MCMC algorithm. The low registration errors and high overlap rates of our algorithm on five datasets prove that the Bayesian model we proposed has not been trapped in local modes. The template reference molecule defined in the Bayesian model is useful in dealing with outliers and missing data. However, the registration error goes up on Skeleton Hand and Dragon dramatically. Due to the anisotropic scale transformation caused by the rough shapes and missing data, the MCMC combined with Bayesian model converges slowly. In the future, we plan to target at anisotropic scale transformation for faster convergent rate.

Although we have just considered pairwise matching of two configurations here, our method has the potential to be extended for matching multiple molecules. And it would be interesting to consider applications that assume nonrigid or even nonlinear transformations between the configurations. Our model allows for such transformations, with suitably adapted implementation.

Acknowledgment

This research was funded by National Natural Science Foundation of China [grant number 61772159], Natural Science Foundation of Shandong Province [grant number

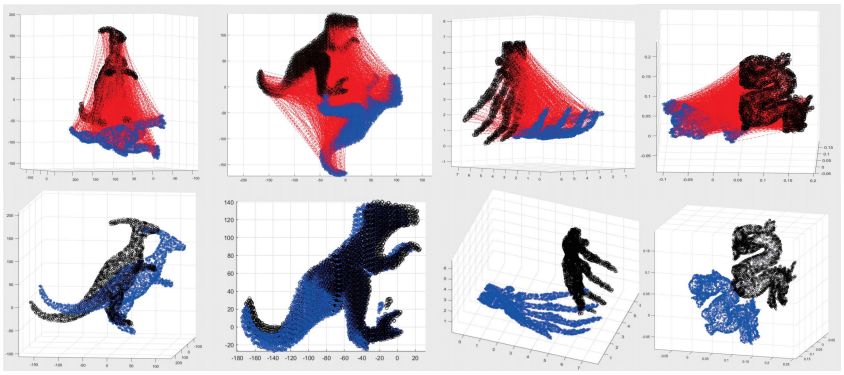


Fig. 6. The matching results and registration results of our method on Parasaurolophus, T-Rex, Skeleton Hand, and Dragon datasets. The first column shows the matching results of our algorithm, and the second column shows the registration results.

ZR2017MF026], and Knowledge Innovation Fund of Harbin Institute of Technology [grant number HIT.NSRIF.201703], in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

References

[1] Zitova, B.; Flusser, J. and Sanroma, G.: Image registration methods: a survey, Image Vis. Comput. vol. 21, no. 11, pp. 977-1000 (2003)

[2] Du, S.; Guo, Y.; Ganroma, G.; *et al.*: Building dynamic population graph for accurate correspondence detection, Medical image analysis, vol. 26, no. 1, 2069-2077 (2015)

[3] Ju, J.; Loew, M.; Ku, B.; *et al.*: Hybrid retinal image registration using mutual information and salient features, IEICE TRANSACTIONS on Information and Systems, vol. 99, no. 6, pp. 1729-1732 (2016)

[4] Shao, X.; Xing, J.;Lv, J.; *et al.*: Unconstrained Face Alignment without Face Detection, Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 2069-2077 (2017)

[5] Zhao, A.; Fu, K.; Wang, S.; *et al.*: Aircraft Recognition Based on Landmark Detection in Remote Sensing Images, IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 8, pp. 1413-1417 (2016)

[6] Ma, Y.;Guo, Y.; Lei, Y.; *et al.*: Efficient rotation estimation for 3D registration and global localization in structured point clouds, Image Vis. Comput. vol. 67, pp. 52-66 (2017)

[7] Zadeh, A.; Baltrusaitis, T. and Morency, L.P.: Convolutional experts constrained local model for facial landmark detection, Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 2051-2059 (2017)

[8] Ye, P. and Liu, F.: Multiple Gaussian mixture models for image registration, IEICE TRANSACTIONS on Information and Systems, vol. 97, no. 7, pp. 1927-1929 (2014)

[9] Green, P.J.: MAD-Bayes matching and alignment for labelled and unlabelled configurations, Geometry Driven Statistics, vol. 121, pp. 377 (2015)

[10] Besl, P.J. and McKay, N.D.: A method for registration of 3D shapes, IEEE Trans. Pattern Anal. Mach. Intell. vol. 14, no. 2, pp. 239-256 (1992)

[11] Chen, Y. and Medioni, G.: Object modelling by registration of multiple range images, Image Vis. Comput. vol. 10, no. 3, pp. 145-155 (1992)
[12] Ying, S.; Peng, J.; Du, S. and Qiao, H.: A scale stretch method based on ICP for 3-D data registration, IEEE Trans. Autom. Sci. Eng. vol. 6, no. 3, pp. 559-565 (2009)
[13] Du, S.; Zheng, N.; Meng, G.; Yuan, Z. and Li, C.: Affine registration of point sets using ICP and ICA, IEEE Signal Process. Lett. vol. 15, pp. 689-692 (2008)
[14] Ying, S.; Peng, J.; Zheng, K. and Qiao, H.: Lie group method for data set registration problem with anisotropic scale deformation, ACTA Autom. Sin. vol. 35, pp. 867-874 (2009)
[15] Zha, H.; Ikuta, M. and Hasegawa, T.: Registration of range images with different scanning resolutions, in: Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, Nashville, TN, USA, pp. 1495-1500 (2000)
[16] Du, S.; Zheng, N.; Ying, S. and Liu, J.: Affine iterative closest point algorithm for point set registration, Pattern Recognit. Lett. vol. 31, pp. 791-799 (2010)
[17] Du, S.; Zheng, N.; Xiong, L.; Ying, S. and Xue, J.: Scaling iterative closest pointalgorithm for registration of m-D point sets, J. Vis. Commun. Image Represent. vol. 21, pp. 442-452 (2010)
[18] Liu, Y.: Constraints for closest point finding, Pattern Recognit. Lett. vol. 29, no. 7, pp. 841-851 (2008)
[19] Liu, Y.: Penalizing closest point sharing for automatic free form shape registration, IEEE Trans. Pattern Anal. Mach. Intell. vol. 33, no. 5, pp. 1058-1064 (2011)
[20] Chetverikov, D.; Stepanov, D. and Krsek, P.: Robust Euclidean alignment of 3D point set: the trimmed iterative closet point algorithm, Image Vis. Comput. vol. 23, no. 3, pp. 299-309 (2005)
[21] Phillips, J.; Ran, L. and Tomasi, C.: Outlier robust ICP for minimizing fractional RMSD, in: Proceedings of Sixth International Conference on 3-D Digital Imaging and Modeling, pp. 427-434 (2007)
[22] Du, S.; Zhu, J.; Zheng, N.; Liu, Y. and Ce, L.: Robust iterative closest point algorithm for registration of point sets with outliers, Opt. Eng. vol. 50, no. 8, pp. 087001 (2011)
[23] Du, S.; Zhu, J.; Zheng, N.; Zhao, J. and Li, C.: Isotropic scaling iterative closest point algorithm for partial registration, Electron. Lett. vol. 47, no. 14, pp. 799-800 (2012)
[24] Dong, J.; Peng, Y.;Ying, S.; *et al.*: LieTrICP: An improvement of trimmed iterative closest point algorithm, Neurocomputing, vol. 140, pp. 67-76 (2014)
[25] Li, W.; Yin, Z.; Huang, Y. and Xiong, Y.: Three-dimensional pointbased shape registration algorithm based on adaptive distance function, Computer Vision, IET, vol. 5, no. 1, pp. 68-76 (2011)
[26] Green, P.J. and Mardia, K.V.: Bayesian Alignment Using Hierarchical Models, With Applications in Protein Bioinformatics, Biometrika, vol. 93, no. 2, pp. 234-254 (2006)
[27] Dryden, I.L.; Hirst, J.D. and Melville, J.L.: Statistical analysis of unlabeled point sets: comparing molecules in cheminformatics, Biometrics, vol. 63, pp. 237-251 (2007)
[28] Mian, A.; Bennamoun, M. and Owens, R.: Three-dimensional modelbased object recognition and segmentation in cluttered scenes, IEEE Trans. Pattern Anal. Mach. Intell. vol. 28, no. 10, pp. 1584-1601 (2006)
[29] Dong, J.; Peng, Y.; Ying, S.; *et al.*: LieTrICP: An improvement of trimmed iterative closest point algorithm, Neurocomputing, vol. 140, pp. 67-76 (2014)