Multiple Kernel Point Set Registration

Thanh Minh Nguyen and Q. M. Jonathan Wu*, *Senior Member, IEEE*

***Abstract*—The finite Gaussian mixture model with kernel correlation is a flexible tool that has recently received attention for point set registration. While there are many algorithms for point set registration presented in the literature, an important issue arising from these studies concerns the mapping of data with nonlinear relationships and the ability to select a suitable kernel. Kernel selection is crucial for effective point set registration. We focus here on multiple kernel point set registration. We make several contributions in this paper. First, each observation is modeled using the Student's t-distribution, which is heavily tailed and more robust than the Gaussian distribution. Second, by automatically adjusting the kernel weights, the proposed method allows us to prune the ineffective kernels. This makes the choice of kernels less crucial. After parameter learning, the kernel saliencies of the irrelevant kernels go to zero. Thus, the choice of kernels is less crucial and it is easy to include other kinds of kernels. Finally, we show empirically that our model outperforms state-of-the-art methods recently proposed in the literature.**

***Index Terms*—Multiple kernel learning, point set registration, Student's t-distribution.**

I. Introduction

POINT set registration plays an important role in the field of medical image analysis and pattern recognition. With the interest features extracted from an image, such as the boundary points, locations of corners, or salient regions, the objective of point set registration is to find the spatial transformation, mapping as well as the correspondence between them. Algorithms operating on the point sets extracted from the input data can help to solve many computer vision problems such as medical image analysis [1]–[3] human pose estimation [4], and body shape modeling [5]. Issues such as noise arising from the processes of image acquisition, high dimensional non-rigid mappings, and the existence of outliers complicate accurate point set registration.

In the last few decades, a number of algorithms have been proposed for point matching. In [6], thin-plate splines are used to define the spatial transformation between corresponding sets of landmarks. This approach has been extended in [7] to take into account landmark localization errors for 3D tomographic images of the human brain. Among them, iterative closest point (ICP) [8], [9] is one of the most well known approaches for point set registration. ICP iteratively assigns correspondence and then finds the least square transformation using the estimated correspondence. The main advantage of the ICP is that it is fast and accurate in many cases. An important drawback of this algorithm is that it lacks robustness with respect to the initial transformation. To improve the convergence behavior of ICP and its robustness against outliers, extensions of ICP have been presented in [10]–[12]. Many researchers have extended ICP to study a number of key problems in medical imaging, such as retinal image registration [13] and multimodal fluorescein angiogram sequence [14]. A graph in [15] represents the local neighborhood relationship. Registration is converted to a graph matching problem. In [16], a non-rigid transformation approach is proposed in a reproducing kernel Hilbert space.

More recent work has drawn upon the rich information-theoretic literature for inspiration on new metrics for point set correspondence. Among them, finite Gaussian mixture model (GMM) based probabilistic registration algorithms has received considerable attention. The main idea is to represent the point sets by a density function. Registration is modeled as a density estimation problem under different transformation models. In [17], thin-plate splines (TPS) is a natural non-rigid extension of the affine map and is used to model the non-rigid transformation. The TPS approach has been shown to have great potential for 3D ultrasound registration [18]. A probabilistic method, coherent point drift (CPD), was introduced in [19], [20]. In this method, the registration is treated as a maximum likelihood estimation problem with motion coherence constraint over the velocity field. In [21] and [22], GMM-based registration is treated as an alignment between two distributions corresponding to two point sets. The study of GMM-based registration has attracted growing attention in medical image registration, such as 3D coronary artery registration [23] and 3D echo image [24]. These methods, illustrated in [17], [19]–[21], and [22] have achieved great success in point set registration. However, they are symmetrical GMM framework in which all Gaussian components are given the same weight. To improve the algorithm's robustness to outliers, finite asymmetrical GMM (AGMM) based probabilistic registration is proposed in [25]. The main advantage of this method is that it gives each Gaussian component a different weight which is related to the feature similarity between the data point and model point. It has proven to be good performance for point sets matching.

While there are many algorithms for point set registration, an important issue arising from these studies concerns mapping data with nonlinear relationships by selecting a suitable kernel.

Manuscript received October 08, 2015; revised December 11, 2015; accepted December 12, 2015. Date of publication December 22, 2015; date of current version May 28, 2016. The work is supported in part by the Natural Sciences and Engineering Research Council of Canada. *Asterisk indicates corresponding author.*

T. M. Nguyen is with the Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON, N9B-3P4 Canada and also with the Biospective, Inc., Montréal, QC, H4P 1K6 Canada (e-mail: nguyen1j@uwindsor.ca)

*Q. M. J. Wu is with the Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON, N9B-3P4 Canada (e-mail: jwu@uwindsor.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMI.2015.2511063

The idea of defining the proposed method stems from the fact that there is not usually a unique kernel Hilbert space for one problem. Here, we consider a point matching problem in which there are several kernel Hilbert spaces, each of which is sufficient to learn the concepts of interest. We make several contributions in this paper. First, each observation is modeled with the Student's t-distribution. Unlike the Gaussian distribution, the Student's t-distribution has an additional parameter, degrees of freedom (ν), which is a robustness tuning parameter. For the particular case of $\nu = 1$, the Student's t-distribution reduces to the Cauchy distribution. When ν tends to infinity, the Student's t-distribution approaches the Gaussian distribution. Hence, it is heavily tailed and more robust than the Gaussian distribution. Second, instead of a single fixed kernel we use multiple kernels. By automatically adjusting the kernel weights, the proposed method allows us to prune the ineffective kernels. For this reason, the choice of kernels is less crucial. Also, including other kinds of kernels is easy. Finally, the proposed method outperforms state-of-the-art methods recently proposed in the literature on various synthetic and real datasets.

This paper is organized into five sections. In Section II, related research is presented. In Section III we describe the details of the proposed method. Section IV presents the experimental results. And Section V provides our conclusions.

II. Related Works

Notations used throughout this paper are as follows. D is the dimension of the point sets. N and M are number of points in two point sets. Θ is the model parameter. We consider the alignment of two point sets as a probability density estimation problem. One point set $\mathrm{x} \in \mathbb{R}^{N\times D}$ is considered as the data points (target point) and the other one $\mathrm{y} \in \mathbb{R}^{M\times D}$ represents the centroids (model point). The density function $p(\mathrm{x}_i|\Theta)$ of the GMM [26] at a given point x_i with M components is given by

$$p(\mathrm{x}_i|\Theta) = \sum_{j=1}^{M} \alpha_{ij}\varphi(\mathrm{x}_i|\Theta_j). \tag{1}$$

As shown in (1), the mixture model has relied on $\varphi(\mathrm{x}_i|\Theta_j)$ to model the underlying distributions. Note that $\varphi(\mathrm{x}_i|\Theta_j)$ can be any kind of distribution. In [17], [19]–[22], [25], $\varphi(\mathrm{x}_i|\Theta_j)$ is a Gaussian distribution $\Phi(\mathrm{x}_i|\mathrm{y}_j^*, \Sigma)$ with its mean y_j^* and covariance Σ

$$\begin{aligned}&\varphi(\mathrm{x}_i|\Theta_j)\\ &= \Phi(\mathrm{x}_i|\mathrm{y}_j^*, \Sigma)\\ &= \frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma|^{1/2}}\exp\left\{-\frac{1}{2}(\mathrm{x}_i - \mathrm{y}_j^*)^T\Sigma^{-1}(\mathrm{x}_i - \mathrm{y}_j^*)\right\}\end{aligned} \tag{2}$$

where $\Sigma = \sigma^2\mathbf{I}$ and $\mathbf{I}$ is the identity matrix. The location of the j-th Gaussian component is given as $\mathrm{y}_j^* = \mathrm{y}_j + \mathcal{C}(\mathrm{y}_j)$. Where $\mathcal{C}(\mathrm{y}_j)$ is the motion model of the point y_j. The $\mathcal{C}$ is written as the linear combination $\mathcal{C}(\mathrm{y}_j) = \sum_{l=1}^{M}\varphi_l\mathcal{K}(\mathrm{y}_j, \mathrm{y}_l)$. The $\Psi \in \mathbb{R}^{M\times D}$ is the parameter, where $\Psi = \{\varphi_{ld}\}$, $l = (1, 2, \ldots, M)$, and $d = (1, 2, \ldots, D)$. The $\mathcal{K} \in \mathbb{R}^{M\times M}$ is a kernel matrix. Each element of this matrix is given as $\mathcal{K}(\mathrm{y}_j, \mathrm{y}_l) = \exp(-0.5\beta\|\mathrm{y}_j - \mathrm{y}_l\|^2)$. The β is a kernel bandwidth controlling the local structure.

In (1), α_{ij} is the mixture coefficient for all components. In [17], [19]–[22], all Gaussian components are given the same weight, $\alpha_{ij} = 1/M$. To improve the algorithm's robustness, AGMM based probabilistic registration is proposed in [25]. In this method, the weights are defined as follows:

$$\alpha_{ij} = \frac{\exp(-\alpha\|\mathrm{des}(\mathrm{x}_i) - \mathrm{des}(\mathrm{y}_j)\|^2)}{\sum_{l=1}^{M}\exp(-\alpha\|\mathrm{des}(\mathrm{x}_i) - \mathrm{des}(\mathrm{y}_l)\|^2)} \tag{3}$$

where $\mathrm{des}(\mathrm{x}_i)$ and $\mathrm{des}(\mathrm{y}_j)$ respectively denote the SIFT features [27] of point x_i and y_j and α is a control parameter. To account for the noises and outliers, an additional uniform distribution is added into the density function [25]. The density function in (1) is rewritten as

$$p(\mathrm{x}_i|\Theta, w) = w\sum_{j=1}^{M}\alpha_{ij}\varphi(\mathrm{x}_i|\Theta_j) + (1-w)p(\mathrm{o}) \tag{4}$$

where the uniform distribution is given as $p(\mathrm{o}) = 1/N$. And the weight of the uniform distribution satisfies the constraints $0 \le w \le 1$. Given the density function $p(\mathrm{x}_i|\Theta, w)$ in (4), the likelihood function is written in the form

$$p(\Theta, w) = \prod_{i=1}^{N} p(\mathrm{x}_i|\Theta, w). \tag{5}$$

To effectively deal with point set registration with more complex transformations, the prior is defined as a Tikhonov regularization

$$p(\mathcal{C}) = \exp(-0.5\lambda\|\mathcal{C}\|_{\mathcal{H}}^2) \tag{6}$$

where $\|\mathcal{C}\|_{\mathcal{H}}^2$ is the norm of $\mathcal{C}(\mathrm{y})$ in the kernel Hilbert space. Given the function $p(\mathrm{x}|\Theta, w)$ in (5) and $p(\mathcal{C})$ in (6), the overall joint distribution is given

$$p(\Lambda|\mathrm{x}) = p(\Theta, w)p(\mathcal{C}). \tag{7}$$

We need to maximize the function $p(\Lambda|\mathrm{x})$ to estimate the parameters $\Lambda = \{\sigma^2, w, \Psi\}$. Please refer to [17], [19]–[22], and [25], for additional parameter estimation details.

III. Proposed Method

A. Objective Function

As shown in (7), to discover nonlinear relationships among two point sets, there is a single kernel space used to map features of the data. In practice, some of the kernel spaces may be useful and can improve the results, while others may be irrelevant for modeling. The idea to define in this part stems from the fact that there is not usually a unique kernel space for one point matching problem.

We consider a set of K kernels $\mathcal{K} = \{\mathcal{K}_1, \mathcal{K}_2, , \ldots, \mathcal{K}_K\}$; $\mathcal{K}_k \in \mathbb{R}^{M\times M}$; $k = (1, 2, \ldots, K)$. Each $\mathcal{K}_k$ is sufficient to learn the concepts of interest. To combine these kernels, we first define kernel saliencies υ_k. In our method, υ_k is the probability that the k-th kernel is relevant. This kernel saliencies υ_k takes a value in the interval [0–1] and satisfies the following constraint $\sum_{k=1}^{K}\upsilon_k = 1$. The kernel is effectively removed from consideration when the kernel saliencies υ_k obtain a value close to zero. It is attractive since it does not require a combinatorial search

over the possible subsets of the kernels which is generally an infeasible task.

Next, to combine these kernels, we define a distribution $\varphi(\mathrm{x}_i|\Theta_j)$ in (4). The distribution $\varphi(\mathrm{x}_i|\Theta_j)$ in our method is given as

$$\varphi(\mathrm{x}_i|\Theta_j) = \sum_{k=1}^{K} \upsilon_k S(\mathrm{x}_i|\mathrm{y}_{jk}^*, \Sigma, \nu) \tag{8}$$

where $S(\mathrm{x}_i|\mathrm{y}_{jk}^*, \Sigma, \nu)$ is the Student's t-distribution

$$S\left(\mathrm{x}_i|\mathrm{y}_{jk}^*, \Sigma, \nu\right) = \frac{\Gamma\left(\frac{\nu}{2}+\frac{D}{2}\right)|\Sigma|^{-1/2}}{(\nu\pi)^{D/2}\Gamma\left(\frac{\nu}{2}\right)} \times \frac{1}{\left[1+\nu^{-1}(\mathrm{x}_i-\mathrm{y}_{jk}^*)^T\Sigma^{-1}(\mathrm{x}_i-\mathrm{y}_{jk}^*)\right]^{(\nu+D)/2}}. \tag{9}$$

In (9), ν is the degree of freedom. The matrix $\Sigma \in \mathbb{R}^{D\times D}$ is the covariance; $\Sigma = \sigma^2\mathbf{I}$. The location of the Student's t-distribution is given as

$$\mathrm{y}_{jk}^* = \mathrm{y}_j + \mathcal{C}_k(\mathrm{y}_j) \tag{10}$$

where $\mathcal{C}_k(\mathrm{y}_j)$ is the motion model of the point corresponding to the k-th kernel. The $\mathcal{C}_k$ is written as $\mathcal{C}_k(\mathrm{y}_j) = \sum_{l=1}^{M}\varphi_{lk}\mathcal{K}_k(\mathrm{y}_j, \mathrm{y}_l)$. The $\Psi_k \in \mathbb{R}^{M\times D}$ is the parameter, where $\Psi_k = \{\varphi_{ldk}\}$, $l = (1, 2, \ldots, M)$, and $d = (1, 2, \ldots, D)$. The $\mathcal{K}_k \in \mathbb{R}^{M\times M}$ is the k-th kernel matrix.

It is worth mentioning that the Cauchy distribution and Gaussian distributions are particular cases for the Student's t-distribution, where $\nu = 1$ and $\nu \to \infty$, respectively. The Student's t-distribution $S(\mathrm{x}_i|\mathrm{y}_{jk}^*, \Sigma, \nu)$ is heavily tailed and more robust than the Gaussian distribution $\Phi(\mathrm{x}_i|\mathrm{y}_j^*, \Sigma)$. Thus, the Student's t-distribution has the flexibility required to fit the shape of the data better than the Gaussian distribution. Given the density function $\varphi(\mathrm{x}_i|\Theta_j)$ in (8), the function in (5) is written in the form

$$\begin{aligned} p(\Theta, w) &= \prod_{i=1}^{N} p(\mathrm{x}_i|\Theta, w) \\ &= \prod_{i=1}^{N}\Big[w\sum_{j=1}^{M}\alpha_{ij}\sum_{k=1}^{K}\upsilon_k S(\mathrm{x}_i|\mathrm{y}_{jk}^*, \Sigma, \nu) + (1-w)p(\mathrm{o})\Big]. \end{aligned} \tag{11}$$

Next, adopting the idea from (6), the prior in our method is defined as

$$p(\mathcal{C}) = \prod_{k=1}^{K} p(\mathcal{C}_k) = \prod_{k=1}^{K}\exp(-0.5\lambda\|\mathcal{C}_k\|_{\mathcal{H}}^2). \tag{12}$$

Given the function $p(\Theta, w)$ in (11) and the prior $p(\mathcal{C})$ in (12), maximizing the overall joint distribution $p(\Lambda|\mathrm{x})$ in (7) is equivalent to maximizing the log-likelihood function

$$\begin{aligned} \log p(\Lambda|\mathrm{x}) &= \log p(\Theta, w) + \log p(\mathcal{C}) \\ &= \sum_{i=1}^{N}\Big[w\sum_{j=1}^{M}\alpha_{ij}\sum_{k=1}^{K}\upsilon_k S(\mathrm{x}_i|\mathrm{y}_{jk}^*, \Sigma, \nu) \\ &\quad + (1-w)p(\mathrm{o})\Big] - \frac{\lambda}{2}\sum_{k=1}^{K}\|\mathcal{C}_k\|_{\mathcal{H}}^2. \end{aligned} \tag{13}$$

Given the log-likelihood function $\log p(\Lambda|\mathrm{x})$ in (13), the next objective is to optimize the parameter set $\Lambda = \{\sigma^2, w, \upsilon_k, \Psi_k, \nu\}$ in order to maximize this log-likelihood function.

B. Parameter Estimation

To estimate the parameters $\Lambda = \{\sigma^2, w, \upsilon_k, \Psi_k, \nu\}$, the EM algorithm [28] is adopted. Each iteration of the EM algorithm consists of two steps. In the E-step, we estimate the missing information with the current parameters. Then the M-step computes the new parameters given the estimate of the missing information

$$\begin{aligned} \text{E step}: &\Im(\Lambda, \Lambda^{(\text{old})}) = \mathbb{E}\left[\log p(\Lambda|\mathrm{x})\right] \\ \text{M step}: &\Lambda^{(\text{new})} = \arg\max_{\Lambda}\Im(\Lambda, \Lambda^{(\text{old})}). \end{aligned} \tag{14}$$

As shown in (A.2), for the E-step the value of the variable z_i is given by using Bayes' theorem which takes the form

$$z_i = \frac{(1-w)}{Nw\sum_{j=1}^{M}\alpha_{ij}\sum_{k=1}^{K}\upsilon_k S(\mathrm{x}_i|\mathrm{y}_{jk}^*, \Sigma, \nu) + (1-w)}. \tag{15}$$

As shown in (A.3), the variable s_{ijk} is given by

$$s_{ijk} = \frac{Nw\alpha_{ij}\upsilon_k S(\mathrm{x}_i|\mathrm{y}_{jk}^*, \Sigma, \nu)}{Nw\sum_{l=1}^{M}\alpha_{il}\sum_{c=1}^{K}\upsilon_c S(\mathrm{x}_i|\mathrm{y}_{lc}^*, \Sigma, \nu) + (1-\omega)}. \tag{16}$$

As shown in (A.10), for the M-step, the solution of $\partial\Im(\Lambda, \Lambda^{(\text{old})})/\partial\sigma^2 = 0$, $\partial Q(\Lambda, \Lambda^{(\text{old})})/\partial w = 0$, and $\partial\Im(\Lambda, \Lambda^{(\text{old})})/\partial\upsilon_k = 0$ [with $\sum_{k=1}^{K}\upsilon_k = 1$] yield the estimates of σ^2, w, and υ_k at the new iteration step

$$\begin{aligned} \sigma^2 &= \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K} s_{ijk}u_{ijk}^{(\text{new})}\|\mathrm{x}_i - \mathrm{y}_{jk}^*\|^2}{D\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K} s_{ijk}} \\ w &= \frac{1}{N}\sum_{i=1}^{N}(1-z_i);\ \upsilon_k = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} s_{ijk}}{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{c=1}^{K} s_{ijc}}. \end{aligned} \tag{17}$$

Setting $\partial\Im(\Lambda, \Lambda^{(\text{old})})/\partial\Psi_k = 0$, we will get

$$\Psi_k = (\text{diag}(\Upsilon_k^T\mathbf{1})\mathcal{K}_k + \lambda\sigma^2\mathbf{I})^{-1}(\Upsilon_k^T\mathrm{x} - \text{diag}(\Upsilon_k^T\mathbf{1})\mathrm{y}). \tag{18}$$

As shown in (A.12), the estimates of the degrees of freedom ν is given by the solution of the equation

$$-\psi\left(\frac{\nu}{2}\right) + \log\left(\frac{\nu}{2}\right) + \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K} s_{ijk}(\log u_{ijk}^{(\text{new})} - u_{ijk}^{(\text{new})})}{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K} s_{ijk}} + \psi\left(\frac{(\nu^{(\text{new})}+D)}{2}\right) - \log\left(\frac{(\nu^{(\text{new})}+D)}{2}\right) + 1 = 0. \tag{19}$$

The various steps of the proposed method are summarized as follows:

Step 1: Step 1: Initialize the parameters $\Lambda = \{\sigma^2, w, \upsilon_k, \Psi_k, \nu\}$: $\sigma^2 = (1/MND)\sum_{i=1}^{N}\sum_{j=1}^{M}||\mathrm{x}_i - \mathrm{y}_j||^2$, $w = 0.7$, $\upsilon_k = 1/K$, $\Psi_k = 0$, and $\nu = 2$.

Step 2: Step 2 (E step): Evaluate the values z_i in (15), s_{ijk} in (16), and $u_{ijk}^{(\mathrm{new})}$ in (A.8) using the current parameter values.

Step 3: Step 3 (M step): Re-estimate the parameters $\Lambda = \{\sigma^2, w, \upsilon_k, \Psi_k, \nu\}$ in (17), (18), and (19).

Step 4: Step 4: Check the convergence of either the log-likelihood function or the parameter. If the convergence criterion is not satisfied, return to step 2.

Once the parameter learning phase is complete, the value of kernel saliencies υ_k is used to score relevant kernels and compare the importance of each kernel. The most important kernel (largest value of υ_k) is chosen to calculate the new location of the component, $\mathrm{y}_j^* = \mathrm{y}_j + \mathcal{C}_{\hat{k}}(\mathrm{y}_j)$, where $\hat{k} = \arg\max_{k=\{1,2\ldots,K\}} \upsilon_k$.

C. Relationship With Existing Methods

Let us try to investigate how the proposed method is related to existing methods. Comparing the mathematical expressions of the log-likelihood function $\log p(\Lambda|\mathrm{x})$ in (13) with that in [19] and [20], we see that if we ignore the term of multiple kernels (set $K = 1$), give the same weights for all components (set $\alpha_{ij} = 1/M$), and use the Gaussian component to model the underlying distributions (set $\nu \to \infty$), these two functions are similar. Comparing the function $\log p(\Lambda|\mathrm{x})$ in (13) with that in [25]. They are similar if we set $K = 1$ and $\nu \to \infty$ for our method. Note that, in this paper, our method automatically adjusts the kernel weights, the proposed method allows us to prune the ineffective kernels. This makes the choice of kernels less crucial. The model with single kernel is presented in [25]. Another advantage of the proposed method compared with [25] is that each observation is modeled with Student's t-distribution, which is heavily tailed and more robust than Gaussian distribution. Therefore, the underlying model of the proposed method is a generalization of the methods in [19], [20], and [25].

IV. Experiments

In order to present our work conveniently, we divide this section into four subsections. In Section IV-A, we validate the design of the proposed method. In Section IV-B, we demonstrate the performance of the proposed method on synthetic data. The results obtained by using our method for real data are presented in Section IV-C. Some discussions of the proposed method will be presented in Section IV-D.

Note that, in Sections IV-B and IV-C, to provide a quantitative comparison, we compare our method to the state-of-the-art algorithms for point set registration: CPD [19], TPS-L2 [21], L2E [16], and AGMM [25]. The CPD, TPS-L2, and L2E methods are implemented using publicly available codes. The mean squared error (MSE) is used to compare the results obtained. This error is the mean Euclidean magnitude distance between calculated and reference landmark positions for the set of validation landmarks. Smaller MSE values indicate better results. In AGMM and our method, we simply choose $\mathrm{des}(\mathrm{x}_i) = \mathrm{x}_i$, $\mathrm{des}(\mathrm{y}_j) = \mathrm{y}_j$, and $\alpha = 0.5$. Unless otherwise specified, 25 kernel features are used for our method. Note that we vary β over [0.1:0.2:5] (Matlab language) to obtain twenty five Gaussian kernels $[\mathcal{K}_k(\mathrm{y}_j, \mathrm{y}_l) = \exp(-0.5\beta\|\mathrm{y}_j - \mathrm{y}_l\|^2)]$. All methods are initialized with the same conditions: Gaussian kernel width ($\beta = 2$), regularization weight ($\lambda = 1$), data normalized to unit variance, and zero mean. Note that the unit of MSE is the error after the normalization. All compared methods are run until convergence of the iteration steps is achieved.

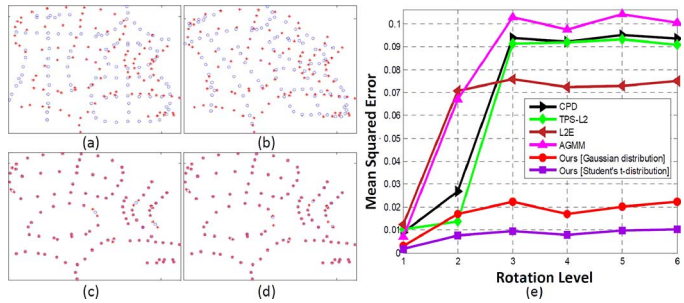

Fig. 1. Gaussian distribution vs. Student's t-distribution. (a) sample data; (b) CPD [MSE = 0.1056]; (c) our method with Gaussian distribution [MSE = 0.0029]; (d) our method with Student's t-distribution [MSE = 0.0017]; (e) average registration error for all 600 samples.

A. Validating the Proposed Method

To evaluate the robustness of the Student's t-distribution compared with Gaussian distribution, a dataset with 600 samples from [17] and [34] (from Table I, Chinese character, Rotation) was used. In this experiment, instead of using the Student's-t distribution as shown in (9), the Gaussian distribution is used. Note that, we set $\nu \to \infty$ for our method. From Fig. 1(b)–(d), we show the results obtained by employing CPD, our method with Gaussian distribution, and our method with Student's t-distribution, respectively. We find that our method based on the Student's t-distribution can obtain a better result. We also compared the performance of all compared methods for all 600 samples in Fig. 1(e). Our method has the lowest average registration error, and demonstrates a better result compared with others.

The experiment shown in Fig. 2 provides more details about the ability of the proposed method in scoring relevant kernels and evaluating the importance of each kernel. The original data is shown in Fig. 2(a). In this experiment, ten kernels features are used for our method. We vary β over [1:1:10] (Matlab language) to obtain ten Gaussian kernels $[\mathcal{K}_k(\mathrm{y}_j, \mathrm{y}_l) = \exp(-0.5\beta\|\mathrm{y}_j - \mathrm{y}_l\|^2)]$. The value of υ_k obtained by the proposed method is shown in Fig. 2(e). Based on this consideration, we can say that kernel #2 is the most important in this experiment. The kernel #6 ranks sixth. And the kernel #10 is the less important one. To clarify this point, from Fig. 2(b) to (d) we show the performance of our method for each kernel [kernel #10, kernel #6, and kernel #2]. As we can see in Fig. 2(f), the MSE value by using kernel #2 is smallest. The kernels in Fig. 2(c) ranks sixth with MSE = 0.0463. Kernel #10 has the greatest value MSE = 0.0681. Clearly, by automatically adjusting the kernel weights, the proposed method allows us to make the choice of

TABLE I
COMPARISON OF THE PROPOSED METHOD WITH OTHER METHODS, (REGISTRATION ERROR: MSE)

Data		CPD [19]	TPS-L2 [21]	L2E [16]	AGMM [25]	Ours
Fish	Deformation [500 samples]	0.0070	0.0094	0.0062	0.0095	0.0041
	Noise [600 samples]	0.0045	0.0080	0.0062	0.0072	0.0042
	Occlusion [600 samples]	0.0565	0.0850	0.0789	0.0753	0.0155
	Outliers [500 samples]	0.0043	0.0076	0.0044	0.0074	0.0043
	Rotation [600 samples]	0.0630	0.0801	0.0533	0.0745	0.0089
Chinese character	Deformation [500 samples]	0.0134	0.0133	0.0124	0.0153	0.0034
	Noise [600 samples]	0.0113	0.0112	0.0115	0.0134	0.0040
	Occlusion [600 samples]	0.0630	0.0770	0.0615	0.0822	0.0142
	Outliers [500 samples]	0.0104	0.0104	0.0119	0.0111	0.0040
	Rotation [600 samples]	0.0684	0.0652	0.0632	0.0799	0.0077
Mean		**0.0302**	**0.0367**	**0.0310**	**0.0376**	**0.0070**

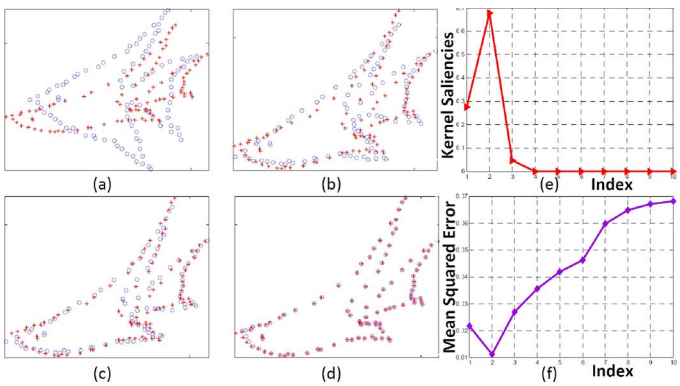

Fig. 2. Kernel comparison: (a) sample data; (b) the tenth kernel [MSE $=$ 0.0681]; (c) the sixth kernel [MSE $=$ 0.0463]; (d) the second kernel [MSE $=$ 0.0112]; (e) the kernel saliencies υ_k; (f) the average registration error of our method for different kernels.

kernels less crucial. And this choice does not require knowledge of any kernels.

B. Synthetic Data

In the first experiment of this subsection, the synthesized datasets from [17] and [34] are used to compare the performance of the proposed algorithm with that of others. This dataset has two shapes: a fish and a Chinese character. For each shape, there are five tests of the data designed to measure robustness: deformation [500 samples], noise [600 samples], occlusion [600 samples], outliers [500 samples], and rotation [600 samples]. In each test, 100 samples are generated for each degradation level. There are 5,600 samples in this synthesized dataset. Fig. 3 shows four of the 5,600 samples. The first row shows the original data. The goal is to align the model blue point set onto the red target point set. The second and last rows of Fig. 3 present the results obtained by employing AGMM and our method, respectively. Looking closely at the marked boxes, we can see that a small number of points have been missed using AGMM. The proposed method, shown in the last row, achieves better accuracy. The registration error in each test (deformation, noise, occlusion, outliers, and rotation) obtained by employing CPD, TPS-L2, L2E, AGMM, and the proposed method, are presented in Table I. As shown in this table, the proposed method has the lowest MSE.

Fig. 4 shows the deformation field. In this experiment, the ground truth is shown in Fig. 4(a) where we want to align the model blue point set onto the red target point set. In Fig. 4(a)–(f), we show the deformation fields obtained by CPD, TPS-L2, L2E, AGMM, and the proposed method. As shown, some points obtained by CPD and TPS-L2 are aligned improperly. The results of CPD and TPS-L2 are less accurate than those of L2E and AGMM. As evident from Fig. 4(f), the proposed method outperforms other methods and has the lowest MSE.

C. Real Data

In the first experiment of this subsection, real 4DCT dataset (4DCT1–4DCT10) obtained from [35], [36] is used to compare the proposed algorithm with CPD, TPS-L2, L2E, and AGMM. Each sample in this dataset has a pair of 3D lung data points selected by experts. Fig. 5(a) shows one example 4DCT6 [slice 110] of this 3D image. And Fig. 5(b) show the pair of data points. The objective in this example is to align the model point set (blue) onto the target point set (red). Fig. 5(c)–(f) shows the results obtained by implementing CPD, TPS-LS, AGMM, and our method. In this experiment, CPD and AMGG in Fig. 5(c) and (e) yield a better result with a lower MSE compared to the TPS-LS method in Fig. 5(d). Compared to these methods, the result of our method is very good with the lowest MSE $=$ 0.0025. We also show the performance of all compared method [CPD, TPS-L2, L2E, and AGMM] for all samples (4DCT1–4DCT10) of this dataset. As shown in Fig. 6, the average MSE obtained by our algorithm is the lowest.

In next experiment, we show the performance of all methods using the real COPDgene dataset [37]. There are 10 samples in this dataset (COPD1–COPD10), an example of which is shown in Fig. 7(a). The pair of data points of COPD8 is shown Fig. 7(b). In Fig. 7(c), we show the performance of our method when we set $\nu \rightarrow \infty$. Note that in this case Student's t-distribution becomes the Gaussian one. A visual inspection indicates that our method with the Student's t-distribution, seen in Fig. 7(d), yields a better result compared against the results in Fig. 7(c). We also show the performance of all compared methods [CPD, TPS-L2, L2E, AGMM, our method with Gaussian distribution, and our method with Student's t-distribution] for all samples of this dataset. As shown in Fig. 7(e), the average MSE obtained by our algorithm is the lowest.

To make all compared methods consistent in comparison to the same data [http://www.dir-lab.com/Results.html], we shown one experiment of the 4DCT6 in Fig. 8. The original data of 4DCT6 with the size of 512x512x128 is shown in Fig. 8(a). The original data is normalized to have a zero mean and unit variance as shown in Fig. 8(b). In Fig. 8(c), the original

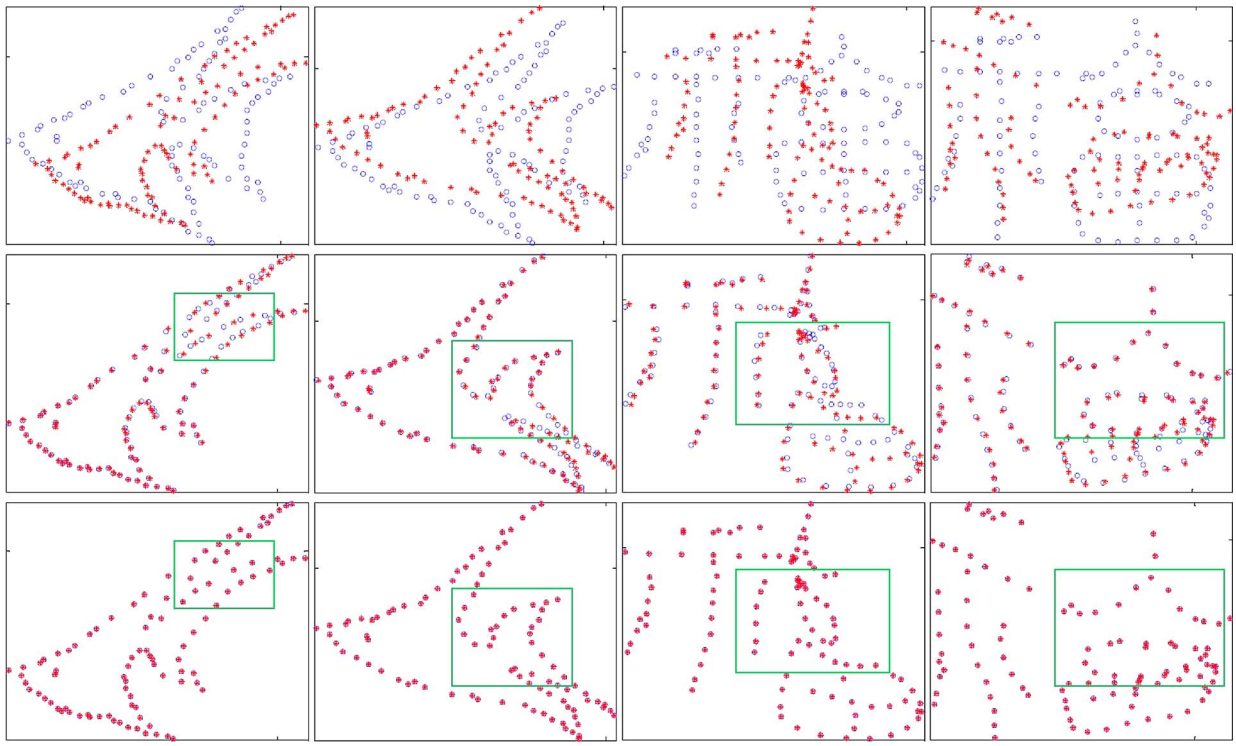

Fig. 3. The goal is to align the model blue point set onto the red target point set. First row: original data; second row: AGMM; last row: our method.

TABLE II
COMPARISON OF THE PROPOSED METHOD WITH OTHERS [UNIT OF MILLIMETRE, NORMALIZED UNIT (NU)]

Data	CPD [19]		TPS-L2 [21]		L2E [16]		AGMM [25]		Ours	
	normalize	mm	normalize	mm	normalize	mm	normalize	mm	normalize	mm
4DCT1	0.009	0.977	0.014	1.45	0.010	1.083	0.009	0.978	0.002	0.19
4DCT2	0.009	0.993	0.016	1.779	0.009	1.091	0.009	0.996	0.002	0.191
4DCT3	0.011	1.252	0.021	2.350	0.012	1.393	0.011	1.257	0.002	0.256
4DCT4	0.015	1.606	0.025	2.618	0.016	1.699	0.015	1.606	0.002	0.226
4DCT5	0.018	1.916	0.026	2.738	0.018	1.934	0.018	1.890	0.002	0.222
4DCT6	0.018	1.871	0.028	2.808	0.019	2.005	0.018	1.883	0.003	0.296
4DCT7	0.014	1.687	0.025	3.081	0.015	1.779	0.014	1.697	0.001	0.194
4DCT8	0.016	1.805	0.043	4.301	0.018	2.020	0.016	1.836	0.002	0.297
4DCT9	0.014	1.482	0.016	1.582	0.015	1.546	0.014	1.488	0.002	0.259
4DCT10	0.017	1.764	0.027	2.795	0.019	1.928	0.017	1.737	0.002	0.248
Mean	**0.014**	**1.535**	**0.024**	**2.550**	**0.015**	**1.648**	**0.014**	**1.537**	**0.002**	**0.238**
COPD1	0.028	2.763	0.053	5.963	0.032	3.366	0.028	2.803	0.003	0.496
COPD2	0.033	4.705	0.051	6.592	0.036	5.06	0.034	4.780	0.002	0.342
COPD3	0.014	1.669	0.016	1.973	0.015	1.824	0.014	1.674	0.002	0.453
COPD4	0.030	3.032	0.051	5.314	0.034	3.993	0.031	3.096	0.002	0.434
COPD5	0.029	2.925	0.045	4.525	0.031	3.391	0.030	2.848	0.003	0.446
COPD6	0.027	2.709	0.037	3.733	0.027	2.866	0.027	2.753	0.002	0.376
COPD7	0.020	2.133	0.026	2.749	0.021	2.357	0.020	2.186	0.003	0.585
COPD8	0.028	3.158	0.035	3.991	0.027	3.213	0.028	3.167	0.002	0.349
COPD9	0.024	2.619	0.033	3.661	0.024	2.772	0.024	2.734	0.002	0.369
COPD10	0.023	3.059	0.034	4.652	0.024	3.419	0.023	3.080	0.003	0.508
Mean	**0.026**	**2.877**	**0.038**	**4.315**	**0.027**	**3.226**	**0.026**	**2.912**	**0.002**	**0.436**

data is shown in units of millimetre [each voxel has a size of $0.97 \times 0.97 \times 2.5$ mm]. For each data set in Table II, the MSE results achieved by all compared approaches are calculated for both units of normalization [as shown in Fig. 8(b)] and units of millimeters [as shown in Fig. 8(c)]. Looking at the Table II, compared with CPD, TPS-L2, L2E, AGMM, and methods in http://www.dir-lab.com/Results.html, our method performs very well in both the 4DCT and COPD dataset.

We use another dataset (Sampled4D) of thoracic 4D CT images [35], [36] to evaluate the performance of the proposed method against the others. There are 10 samples [Case1Pack-Case10Pack]. Each sample consists of six expiratory phase (T00, T10, T20, T30, T40 and T50). In this experiment, T00 is assigned as the target point. T10, T20, T30, and T40 are assigned as the models. In total, 50 pairs of 3D lung data points are used in this experiment. As evident from the results in III ,

TABLE III
COMPARISON OF THE PROPOSED METHOD WITH OTHER METHODS FOR THE SAMPLED4D [4DCT] DATASET, (THE REGISTRATION ERROR: MSE)

Data		CPD [19]	TPS-L2 [21]	L2E [16]	AGMM [25]	Ours [Gaussian distribution]	Ours [Student's t-distribution]
Sampled4D [Case1Pack –Case10Pack]	T00 vs. T10	0.0063	0.009	0.0071	0.0065	0.0009	0.0008
	T00 vs. T20	0.0098	0.0166	0.0107	0.0101	0.001	0.0008
	T00 vs. T30	0.0114	0.0206	0.0125	0.0118	0.0009	0.0006
	T00 vs. T40	0.0128	0.0238	0.0136	0.0131	0.0009	0.0007
	T00 vs. T50	0.0128	0.0247	0.0136	0.0129	0.0011	0.0008
mean		**0.0106**	**0.0189**	**0.0115**	**0.0109**	**0.0010**	**0.0007**

TABLE IV
COMPARISON OF THE PROPOSED METHOD WITH OTHER METHODS [WITH DIFFERENT LEVELS OF TRAINING AND TESTING DATA] FOR THE COPDGENE DATASET [COPD1-COPD10] (REGISTRATION ERROR: MSE)

Data		Training					Testing [MSEx10^{-2}]				
Training	Testing	CPD [19]	TPS-L2 [21]	L2E [16]	AGMM [25]	Ours	CPD [19]	TPS-L2 [21]	L2E [16]	AGMM [25]	Ours
50%	50%	0.029	0.059	0.032	0.029	0.003	31.6	31.76	31.62	31.61	30.87
55%	45%	0.032	0.051	0.038	0.033	0.003	20.42	20.58	20.41	20.4	20.74
60%	40%	0.014	0.017	0.015	0.014	0.002	11.78	11.75	11.8	11.79	11.62
65%	35%	0.035	0.054	0.036	0.035	0.003	37.53	37.74	37.77	37.66	36.89
70%	30%	0.031	0.048	0.033	0.032	0.002	28.84	29.03	28.82	28.84	28.69
75%	25%	0.026	0.039	0.028	0.027	0.002	20.83	20.83	20.83	20.81	20.66
80%	20%	0.02	0.027	0.022	0.02	0.002	18.91	18.85	18.86	18.87	18.45
85%	15%	0.029	0.037	0.028	0.029	0.002	21.2	21.12	21.21	21.21	20.98
90%	10%	0.024	0.031	0.025	0.025	0.003	19.27	19.72	19.28	19.28	19.02
95%	5%	0.025	0.036	0.025	0.025	0.001	25.49	25.52	25.48	25.47	25.16
mean		**0.026**	**0.040**	**0.028**	**0.027**	**0.002**	**23.59**	**23.69**	**23.61**	**23.59**	**23.31**

that the proposed method, on average, has the lowest MSE (at 0.0007) and outperforms the other methods.

To verify that the results achieved by different approaches are statistically significant, we present the experiment in Table IV. The real COPDgene dataset (10 samples: COPD1–COPD10) is used in this experiment. There are 300 pairs of 3D lung data points that are manually annotated by medical experts for each sample. We used the expiration (eBH) scan as a template and the inspiration (iBH) scan as reference. For each sample, we randomly assign these 300 points to training and testing sets. We then train the first set. After training, we adopt the Shepard's interpolation for the expiration (eBH) of the testing set to construct new data points. Then, we compute the registration error between these data points and their corresponding template landmarks. The average MSE obtained by all compared methods with different levels of training (50%, 55%, 60%, 65%, 70%, 75%, 80%, 95%, 90%, 95%) and testing data are shown in Table IV. Note that all compared methods are repeated in 20 random runs, and the average MSE for all samples (COPD1-COPD10) is recorded. In this table, we see that the MSE values of our method are better than that of other algorithms. To verify that this goodness is statistically significant, we report the p-values, called one tailed paired t-test [38], for our method and others for both training and testing data in Table V. Four groups, corresponding to the four compared algorithms (CPD, TPS-L2, L2E, AGMM), have been created for each data set. Each group in this table reports the p-values produced to compare the registration error (MSE) of our method against other methods. As shown in this table, the p-values are less than 0.05 (5% significance level). For example, the p-value between the algorithms in our method and CPD for the case of 50% training data is 6.40E-17, which is very small. This indicates that the results achieved by different approaches are statistically significant and have not occurred by chance.

To validate the proposed method, our experiment is shown in Fig. 9. In this experiment, the provided landmarks on COPDgene data are used as the ground truth to evaluate the registration accuracy rather than directly working on them. For each of the 10 samples (COPD1–COPD10) of the COPDgene dataset, we segment the left and right lungs using the technique developed in [39]. In Fig. 9(a)–(b), we show the original data (COPD3_iBHCT, slice 50) and the segmentation results obtained from [35]. We then adopt the SIFT features [27] with PeakThresh $= 1\text{E} - 5$ for each slice of the segmented images to extract the landmarks. In Fig. 9(c), we show the landmarks obtained from [27]. We use these landmarks to train all methods. After training, similar to the experiment in Table IV, we adopt Shepard's interpolation to construct new data points for all 300 points for the expiration (eBH). We then compute the registration error between these data points and their corresponding template landmarks (iBH). As shown in Fig. 9(e) and (f), the proposed method performs well for both training and testing data sets.

Fig. 10 presents another experiment to compare the proposed algorithm with CPD, TPS-L2, L2E, and AGMM. Similar to the experiment in Fig. 9, we use the same setting for all compared methods in this experiment. The only difference is that we use the 4DCT dataset (4DCT1–4DCT10). As shown in Fig. 10(e) and (f), the result of our method is very good and the average MSE obtained by our algorithm is the lowest for both testing and training data.

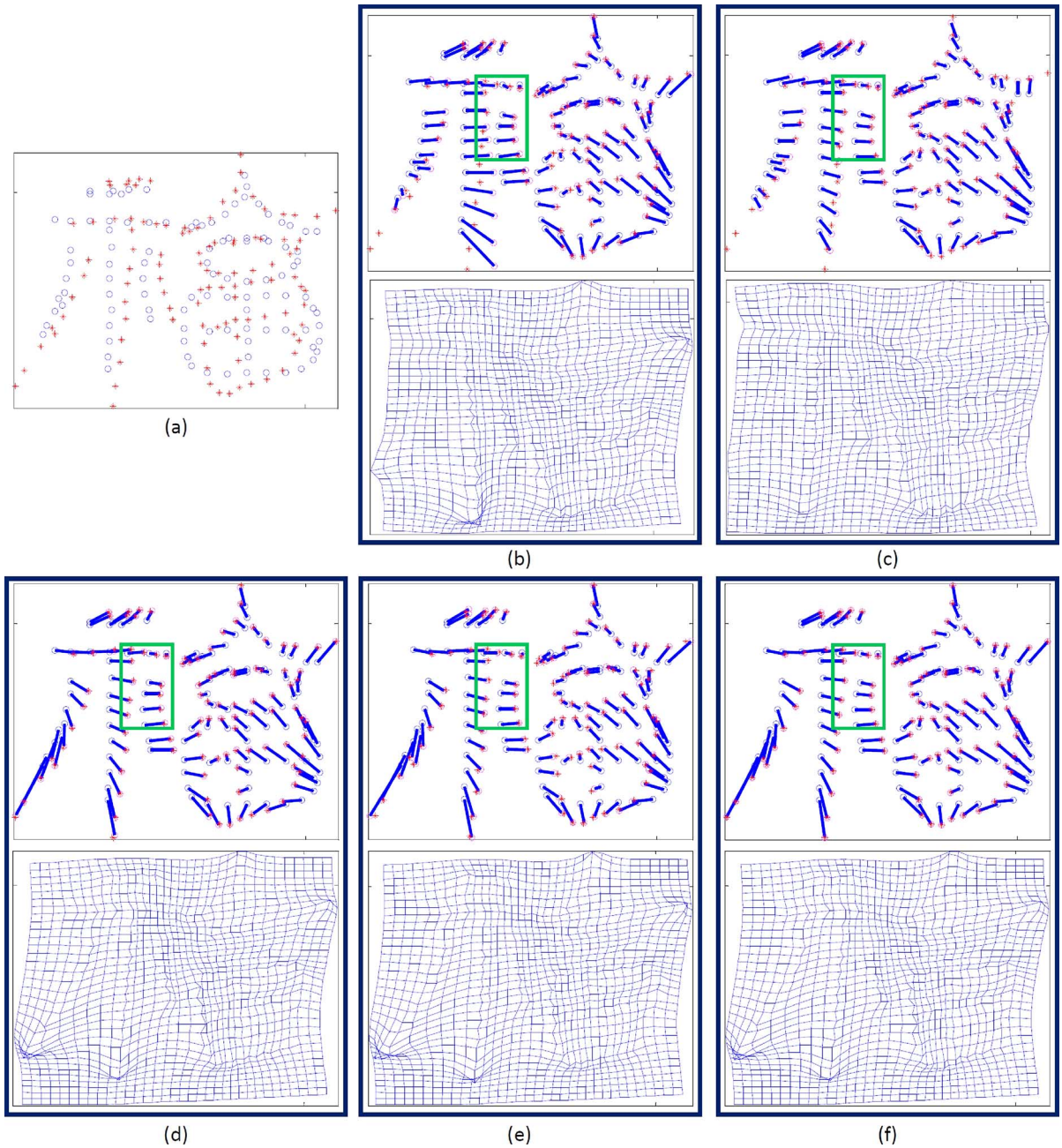

Fig. 4. The deformation field: (a) original data; (b) CPD [MSE = 0.0319]; (c) TPS-L2 [MSE = 0.0247]; (d) L2E [MSE = 0.0091]; (e) AGMM [MSE = 0.0179]; (f) Our method [MSE = 0.0003].

D. Discussion

As mentioned in Section I, an advantage of the proposed method is that including other kinds of kernels is easy and will lead to even better performances. This subsection will show this advantage. To explain the implementation of the proposed method for different types of kernels [Gaussian kernels, polynomial kernels, inverse multiquadric kernels], we present the results of an experiment in Fig. 11. The fist column of this figure includes two examples. The second column of Fig. 11 presents the results obtained by employing our method with Gaussian kernels. In this case, 25 kernel features are used for our method by varying β over [0.1:0.2:5] to obtain 25 Gaussian kernels $[\mathcal{K}_k(\mathrm{y}_j, \mathrm{y}_l) = \exp(-0.5\beta\|\mathrm{y}_j - \mathrm{y}_l\|^2)]$. In the third column of Fig. 11, we use the same conditions as in the second column. The only difference between them is that we use the polynomial kernel in third column of Fig. 11. In this case, 10 kernels are used for our method by varying β over [1:1:10] to obtain 10 polynomial kernels $[\mathcal{K}_k(\mathrm{y}_j, \mathrm{y}_l) = (\mathrm{y}_j \mathrm{y}_l^T + 1)^\beta]$. Similarly, we use inverse multiquadric kernels in the fourth column of Fig. 11 by varying β over [0.01:0.01:0.1] to obtain 10 inverse multiquadric kernels $[\mathcal{K}_k(\mathrm{y}_j, \mathrm{y}_l) = 1/\sqrt{\|\mathrm{y}_j - \mathrm{y}_l\|^2 + \beta}]$. In the last column, we combine these kernels. In this case, we use 45 kernels

$$[25 \text{ Gaussian kernels} + 10 \text{ polynomial kernels} + 10 \text{ inverse multiquadric kernels}]$$

TABLE V
THE p–VALUES OF OUR METHOD AND OTHERS FOR THE TRAINING AND TESTING DATA IN TABLE IV

Data		Training				Testing [MSEx10^{-2}]			
Training	Testing	CPD [19]	TPS-L2 [21]	L2E [16]	AGMM [25]	CPD [19]	TPS-L2 [21]	L2E [16]	AGMM [25]
50%	50%	6.40E-17	4.17E-11	2.19E-17	9.33E-17	3.54E-07	4.62E-03	1.02E-07	1.54E-07
55%	45%	1.78E-16	8.32E-13	8.40E-16	3.63E-17	1.47E-07	7.59E-03	2.11E-07	4.14E-07
60%	40%	6.78E-20	3.10E-12	3.86E-18	1.89E-18	3.36E-07	1.61E-03	5.12E-08	1.60E-07
65%	35%	2.42E-18	3.55E-11	1.98E-17	8.53E-20	3.48E-07	7.61E-05	3.20E-07	4.53E-07
70%	30%	2.71E-17	5.59E-12	5.81E-17	2.85E-17	1.10E-09	1.44E-08	8.21E-10	1.77E-09
75%	25%	1.92E-19	1.23E-12	1.29E-16	3.12E-19	4.04E-08	1.38E-06	2.55E-08	2.53E-08
80%	20%	1.31E-20	7.55E-14	5.12E-19	5.66E-20	4.19E-10	6.18E-09	9.22E-12	4.86E-11
85%	15%	6.89E-20	1.04E-17	1.67E-20	1.30E-19	6.60E-10	4.03E-10	3.27E-10	6.25E-10
90%	10%	1.85E-20	1.28E-18	2.50E-20	1.90E-20	6.31E-09	4.42E-08	8.01E-09	6.02E-09
95%	5%	1.58E-22	2.64E-17	1.14E-21	6.37E-22	1.49E-07	6.50E-07	9.89E-08	1.51E-07

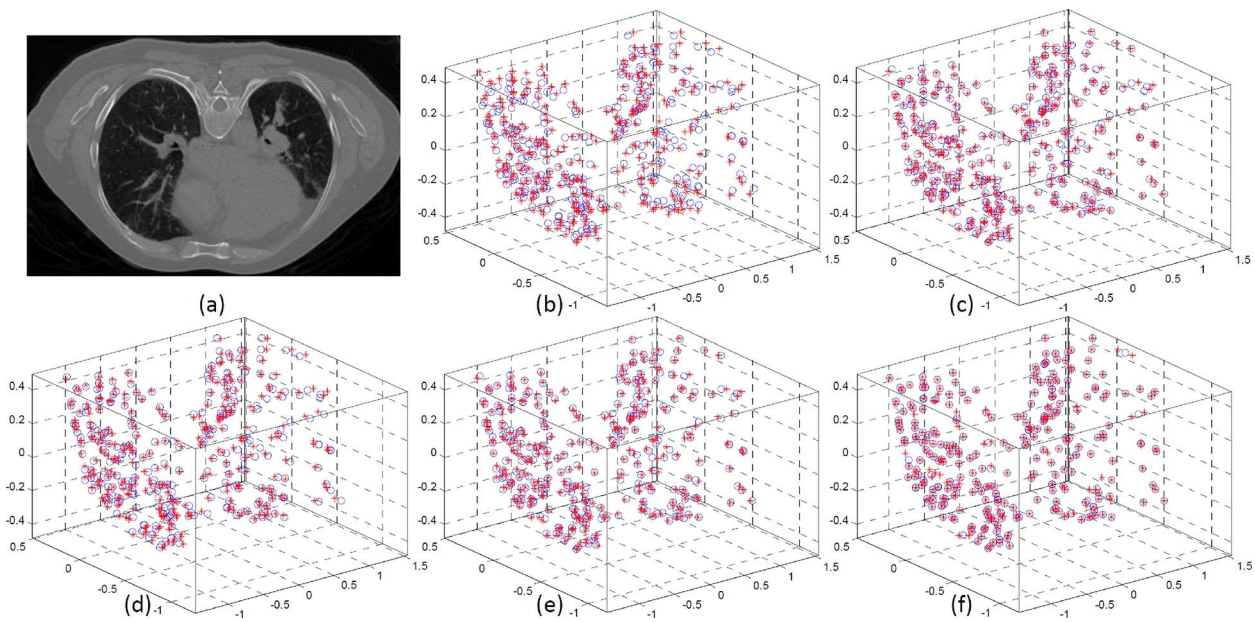

Fig. 5. (a) 4DCT6 [slice 70]; (b) original data; (c) CPD [MSE $=$ 0.0179]; (d) TPS-LS [MSE $=$ 0.0284]; (e) AGMM [MSE $=$ 0.0180]; (f) our method [MSE $=$ 0.0025].

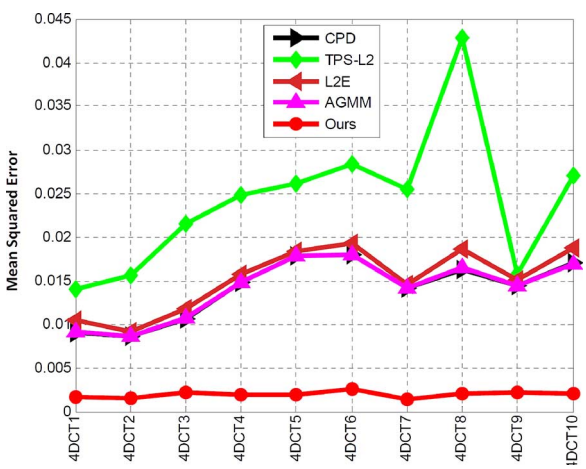

Fig. 6. Comparison of the proposed method with other methods for 4DCT dataset [4DCT1–4DCT10].

for our method. As we can see in the first example (first row of Fig. 11), compared with the polynomial kernels and inverse multiquadric kernels, the Gaussian kernels reduce the error significantly. The proposed method with all kernels [Gaussian $+$ polynomial $+$ inverse multiquadric] in the last row can automatically determine the kernel saliency with MSE $= 5.1\text{E}-4$. Similarly, for the second example (second row of Fig. 11), the inverse multiquadric kernels reduce the error significantly, with MSE $= 6.9\text{E}-4$, compared to the Gaussian and polynomial kernels. The proposed method with all kernels in the last row also determines the kernel saliency with MSE $= 6.9\text{E}-4$. We also compare the performance of these methods for all 500 samples in Table I [Chinese character: Deformation]. As shown in Table VI, the proposed method with all kernels [Gaussian $+$ polynomial $+$ inverse multiquadric] has the lowest average registration error and demonstrates a better result compared to the others.

In medical image registration, the pair of landmarks is hard to obtain due to the large anatomical variation and imaging quality. In the experiment of Table II, we base the validation of our method on the same landmarks as those that are used to drive the registration. Unfortunately, these landmarks are usually unavailable for practical applications. The selection of re-

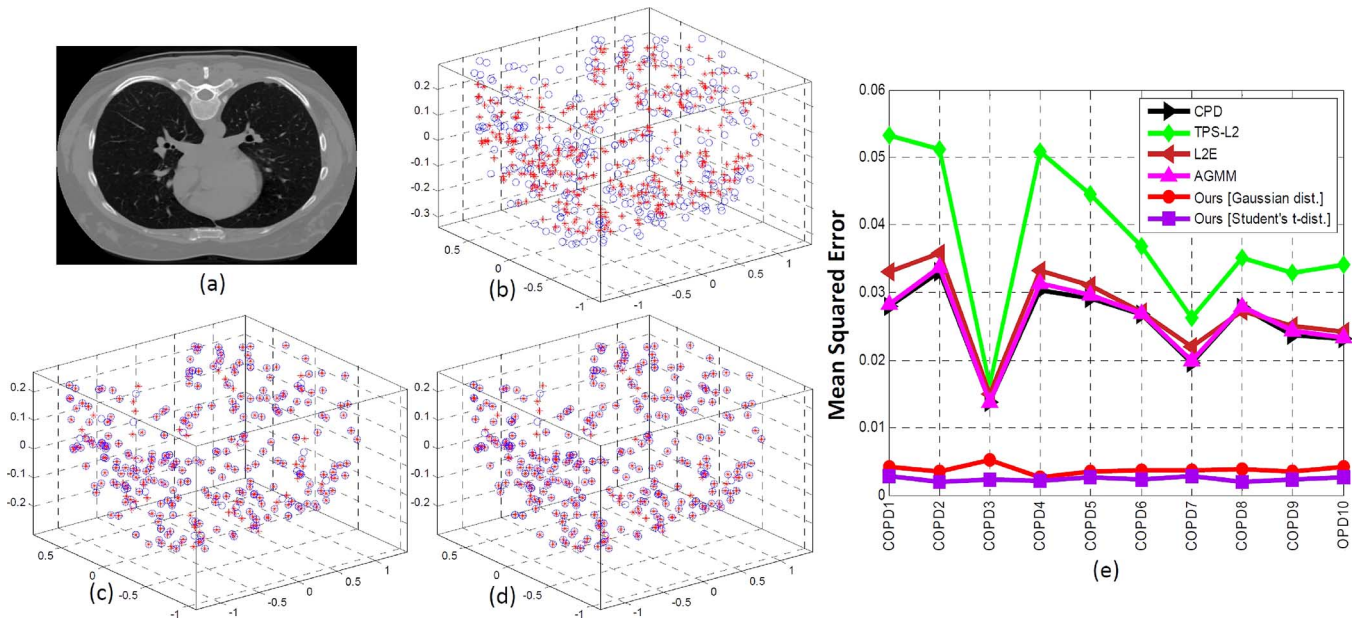

Fig. 7. (a) COPD8_iBHCT [slice 60]; (b) original data; (c) our method with Gaussian distribution [MSE $= 0.0038$]; (d) our method with Student's t-distribution [MSE $= 0.0020$]; (e) the average registration error for COPDgene dataset [COPD1–COPD10].

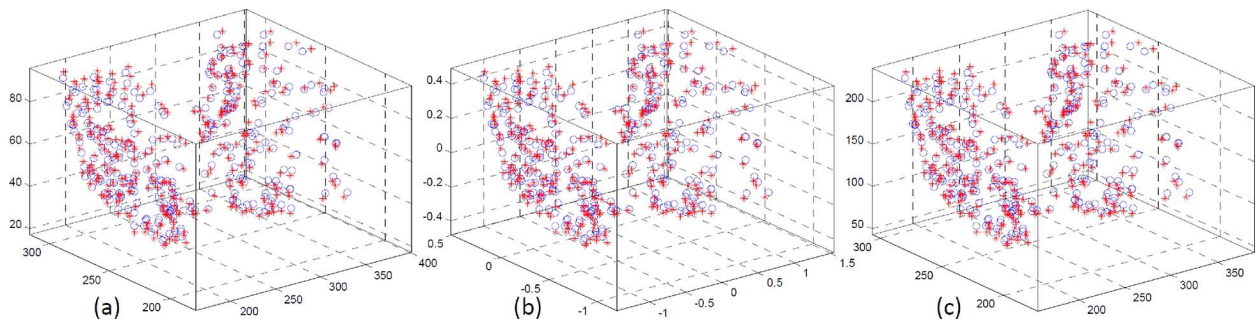

Fig. 8. Explanation of the unit of MSE: (a) original data [pixel unit]; (b) normalized data [normalized (a) to have a zero mean and unit variance]; (c) unit of millimetre.

liable landmark pair is an indispensable step in the point-set method for medical image registration. There are already many studies [2], [3] that tackle the above issue. One solution to select the landmark pair is shown in Fig. 12. Two images (T00 and T50) of the 4DCT1 data are shown. To select the landmark pairs from these two images, we first adopt the method in [24] to complete the registration between T00 and T50 images. The registered image is shown in Fig. 12(c). The composite of T00 and T50 (before registration) is shown in Fig. 12(d). The result after the registration step is shown in Fig. 12(d). Second, from Fig. 12(e), we generate the landmark pairs as shown in Fig. 12(f). In this case, grid coordinates with step 20 are used. Note that, the manually annotated landmarks in Table II are not used as the input point sets for registration; they are used as the ground truth for evaluating the registration accuracy. As shown in Table VIII, the proposed method outperforms other methods for both training and evaluation.

V. Conclusions

In this paper, we present a model for point set registration, focusing on multiple kernel point set registration. The advantage of our method is that it has an ability to score relevant kernels. By automatically adjusting the kernel weights, the proposed method allows us to prune the ineffective kernels and evaluate the importance of each kernel. In addition, it avoids combinatorial searches and is intuitively appealing. We model each observation with the Student's t-distribution, which is heavily tailed and more robust than the Gaussian distribution. We adopt the EM algorithm to maximize the data likelihood and to optimize the parameters. The proposed method has been tested on various datasets. We demonstrate through extensive simulations that the proposed model is superior to state-of-the-art methods recently proposed in the literature for point set registration.

Another possible extension of this work is to study the remaining parameter λ to improve the performance of the proposed method. In the context of this paper, the user sets this parameter based on prior knowledge. A possible solution to overcome this problem is to set this parameter by cross-validation. Identifying a remedy for this problem remains the subject of our research.

Appendix

To estimate the parameters $\Lambda = \{\sigma^2, w, \upsilon_k, \Psi_k, \nu\}$, the EM algorithm [28] is adopted. Applying (14) to (13), following [26],

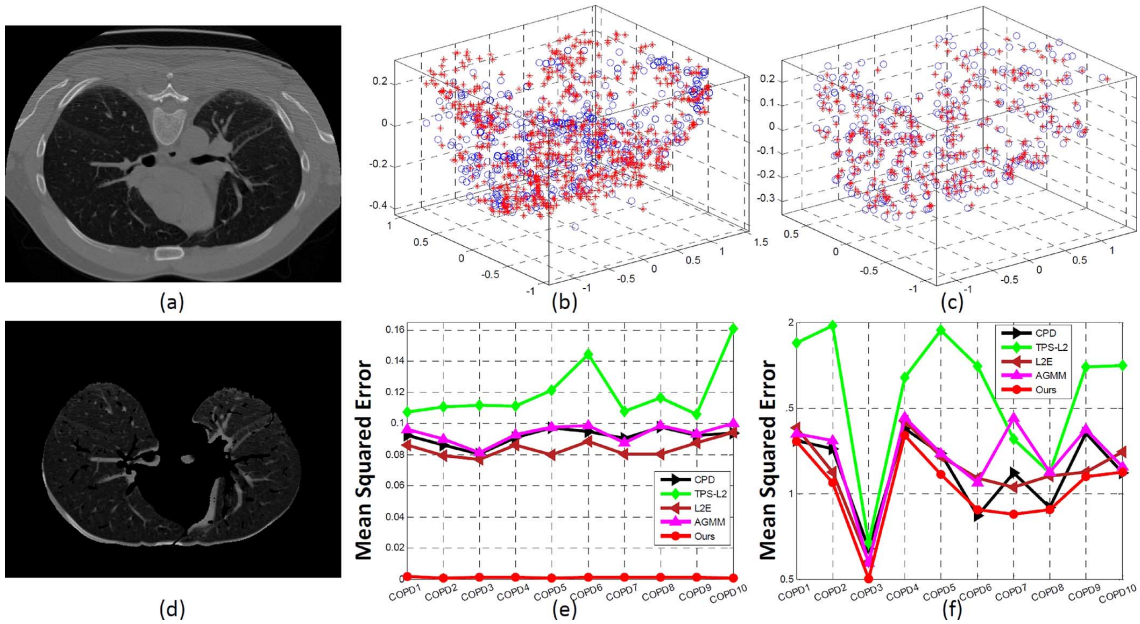

Fig. 9. (a) COPD3_iBHCT [slice 50]; (d) Lung segmentation from [39]; (b) landmarks obtained from [27]; (c) 300 pairs of 3D lung data points for testing data; (e) the average MSE of our method and others for training data; (f) the average MSE of our method and others for testing data.

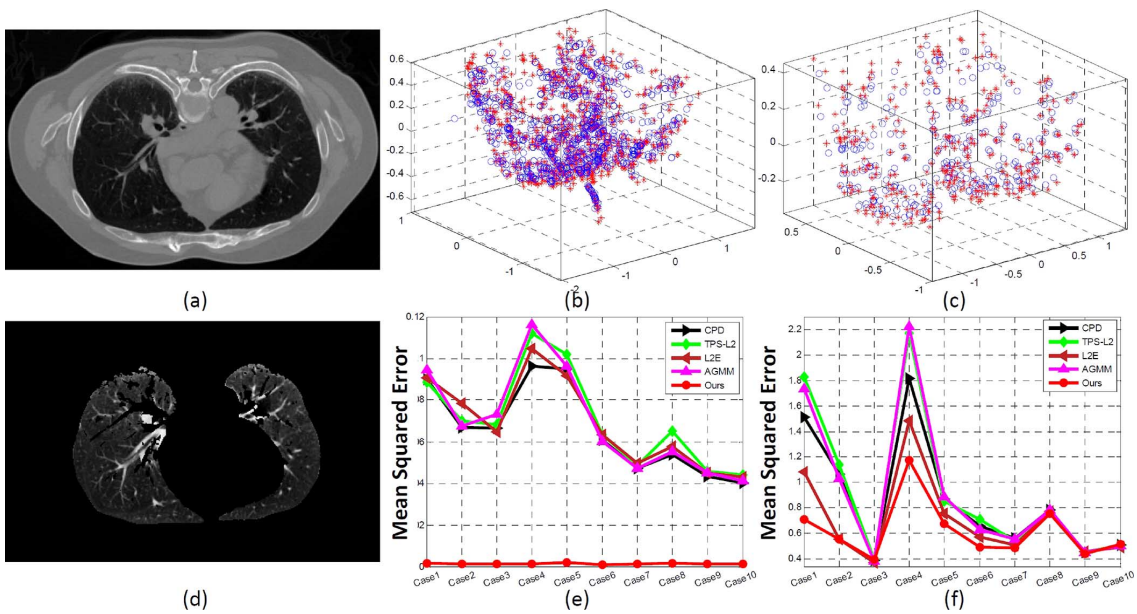

Fig. 10. (a) 4DCT7 [case7_T00, slice 70]; (d) lung segmentation from [39]; (b) landmarks obtained from [27]; (c) 300 pairs of 3D lung data points for testing data; (e) the average MSE of our method and others for testing data.

[29], and omitting terms that are independent of Λ, we obtain the objective function as follow:

$$\begin{aligned}\Im(\Lambda, \Lambda^{(\text{old})}) = &\sum_{i=1}^{N}(1-z_i)\log w + \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K} s_{ijk}\log \upsilon_k \\ &+ \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K} s_{ijk}\mathbb{E}[\log S(\mathrm{x}_i|\mathrm{y}^*_{jk}, \Sigma, \nu)] \\ &+ \sum_{i=1}^{N} z_i \log(1-w) - \frac{\lambda}{2}\sum_{k=1}^{K}\mathrm{trace}(\Psi_k^T \mathcal{K}_k \Psi_k)\end{aligned} \tag{A.1}$$

where the value of the variable z_i is given by using Bayes' theorem which takes the form

$$z_i = \frac{(1-w)}{Nw\sum_{j=1}^{M}\alpha_{ij}\sum_{k=1}^{K}\upsilon_k S(\mathrm{x}_i|\mathrm{y}^*_{jk}, \Sigma, \nu) + (1-w)}. \tag{A.2}$$

The variable s_{ijk} is given by

$$s_{ijk} = \frac{Nw\alpha_{ij}\upsilon_k S(\mathrm{x}_i|\mathrm{y}^*_{jk}, \Sigma, \nu)}{Nw\sum_{l=1}^{M}\alpha_{il}\sum_{c=1}^{K}\upsilon_c S(\mathrm{x}_i|\mathrm{y}^*_{lc}, \Sigma, \nu) + (1-\omega)}. \tag{A.3}$$

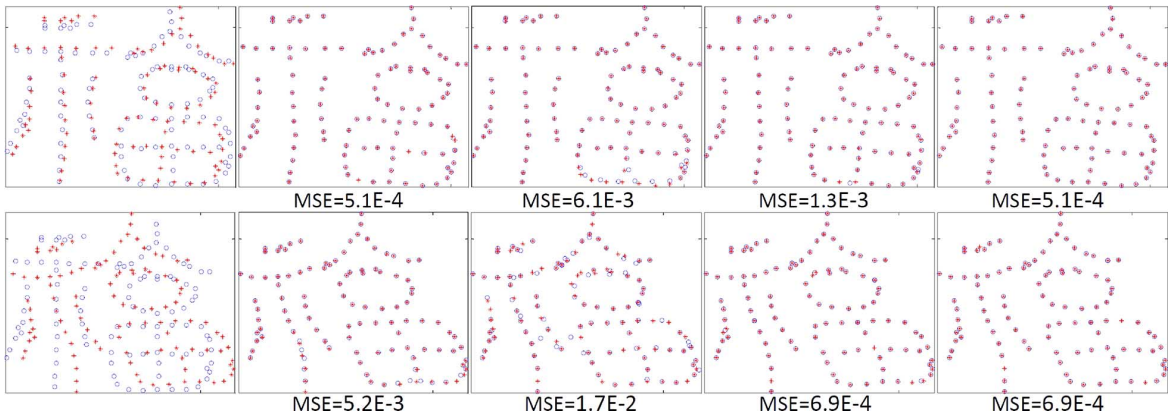

Fig. 11. Different types of kernels: (first column) sample data; (second column) our method with Gaussian Kernels; (third column) our method with Polynomial kernels; (fourth column) our method with Inverse Multiquadric kernels; (last column) our method with all $(\text{Gaussian} + \text{polynomial} + \text{Inverse Multiquadric})$ kernels.

TABLE VI
COMPARISON OF THE PROPOSED METHOD WITH DIFFERENT TYPES OF KERNELS FOR THE FISH DEFORMATION DATASET (500 SAMPLES) (REGISTRATION ERROR: MSE)

CPD [19]	TPS-L2 [21]	L2E [16]	AGMM [25]	Ours [Gaussian kernels]	Ours [polynomial kernels]	Ours [inverse multiquadric kernels]	Ours [all kernels]
0.0070	0.0094	0.0062	0.0095	0.0041	0.011	0.0014	0.0011

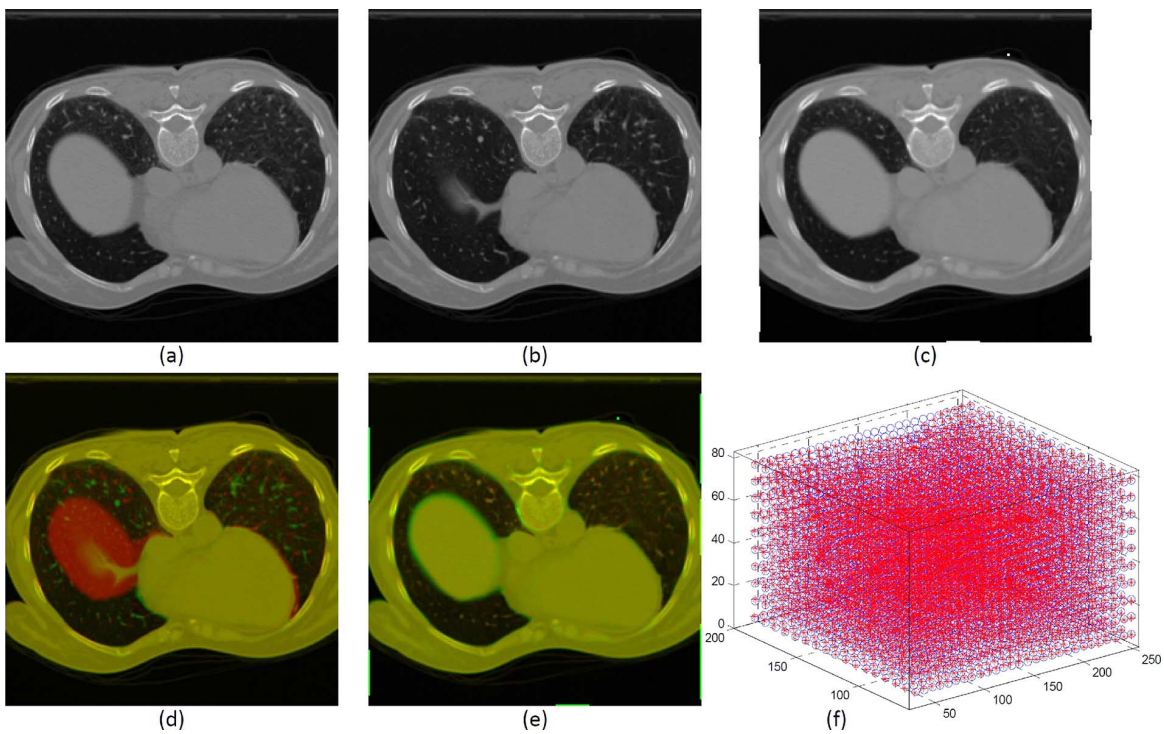

Fig. 12. 4DCT1 data: (a) Fixing image [T00, slice 70]; (b) Moving image [T50, slice 70]; (c) registered image [slice 70]; (d) Composite of two image (a) and (b); (e) Composite of two image (a) and (c); (f) Landmarks obtained from (e).

In (A.1), to estimate the parameter Λ, we need to calculate the expected value of the log Student's t-distribution $\mathbb{E}[\log S(\mathrm{x}_i|\mathrm{y}^*_{jk}, \Sigma, \nu)]$. Note that there is no closed-form solution for maximizing the log-likelihood under a Student's t-distribution. To overcome this problem, the Student's t-distribution in previous models [29]–[33] is represented as a Gaussian distribution with scaled precision

$$S(\mathrm{x}_i|\mathrm{y}^*_{jk}, \Sigma, \nu) \sim \Phi\left(\mathrm{x}_i|\frac{\mathrm{y}^*_{jk}, \Sigma}{u_{ijk}}\right) \mathcal{G}\left(u_{ijk}|\frac{\nu}{2}, \frac{\nu}{2}\right) \quad \text{(A.4)}$$

where $\Phi(\mathrm{x}_i|\mathrm{y}^*_{jk}, \Sigma/u_{ijk})$ is a Gaussian distribution

$$\Phi\left(\mathrm{x}_i|\frac{\mathrm{y}^*_{jk}, \Sigma}{u_{ijk}}\right) = \frac{1}{(2\pi)^{D/2}} \frac{u_{ijk}^{D/2}}{|\Sigma|^{1/2}} \times \exp\left\{-\frac{1}{2} u_{ijk} (\mathrm{x}_i - \mathrm{y}^*_{jk})^T \Sigma^{-1} (\mathrm{x}_i - \mathrm{y}^*_{jk})\right\} \quad \text{(A.5)}$$

and the Gamma distribution $\mathcal{G}(u_{ijk}|\nu/2, \nu/2)$ is given by

$$\mathcal{G}\left(u_{ijk}|\frac{\nu}{2}, \frac{\nu}{2}\right) = \frac{1}{\Gamma(\frac{\nu}{2})} \left(\frac{\nu}{2}\right)^{(\nu/2)} (u_{ijk})^{\nu/2-1} e^{-\nu u_{ijk}/2}. \quad \text{(A.6)}$$

TABLE VII
COMPARISON OF THE PROPOSED METHOD WITH DIFFERENT METHODS [UNIT OF MILLIMETERS]

Data	CPD [19]		TPS-L2 [21]		L2E [16]		AGMM [25]		Ours	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing	Training	Testing
4DCT1	0.011	5.445	0.017	5.540	0.013	5.438	0.010	5.437	0.009	5.243
4DCT2	0.010	5.826	0.014	5.934	0.012	5.815	0.013	5.811	0.009	5.119
4DCT3	0.013	5.769	0.014	5.888	0.011	5.772	0.014	5.758	0.009	5.284
4DCT4	0.012	5.248	0.020	5.304	0.015	5.246	0.014	5.238	0.010	5.216
4DCT5	0.016	5.266	0.021	5.325	0.018	5.233	0.015	5.232	0.013	5.273
4DCT6	0.014	4.899	0.024	5.213	0.016	4.901	0.013	4.885	0.012	4.517
4DCT7	0.014	4.849	0.021	5.059	0.016	4.834	0.014	4.818	0.012	4.766
4DCT8	0.011	5.826	0.014	5.934	0.012	5.815	0.012	5.811	0.009	5.219
4DCT9	0.014	4.899	0.024	5.213	0.016	4.901	0.013	4.885	0.011	4.517
4DCT10	0.016	5.021	0.025	4.669	0.019	4.939	0.015	5.062	0.015	5.001
Mean	**0.013**	**5.305**	**0.019**	**5.408**	**0.015**	**5.289**	**0.013**	**5.294**	**0.011**	**5.016**
COPD1	0.075	14.551	0.068	13.803	0.059	13.718	0.052	13.094	0.049	13.504
COPD2	0.061	12.427	0.051	12.064	0.048	12.098	0.043	11.975	0.042	11.905
COPD3	0.080	11.080	0.091	13.182	0.072	13.216	0.084	12.081	0.082	12.656
COPD4	0.070	11.845	0.071	12.160	0.062	12.170	0.053	11.986	0.051	11.577
COPD5	0.084	11.408	0.145	12.207	0.078	13.110	0.079	11.312	0.073	11.229
COPD6	0.058	12.787	0.067	12.311	0.051	12.581	0.046	12.221	0.046	12.099
COPD7	0.032	10.919	0.022	11.032	0.024	11.070	0.022	11.034	0.022	10.687
COPD8	0.068	10.763	0.062	10.895	0.063	11.670	0.064	10.754	0.059	10.722
COPD9	0.056	11.939	0.055	12.294	0.057	12.410	0.052	12.316	0.049	11.538
COPD10	0.100	17.133	0.0911	17.018	0.073	17.331	0.085	16.284	0.081	16.167
Mean	**0.068**	**12.485**	**0.072**	**12.697**	**0.059**	**12.937**	**0.058**	**12.306**	**0.055**	**12.208**

In (A.6), $\Gamma(\cdot)$ is the Gamma function. From (A.4) and following [29]–[31], the expected value of the log Student's t-distribution $\mathbb{E}[\log S(\mathrm{x}_i|\mathrm{y}^*_{jk},\Sigma,\nu)]$ is given as

$$\begin{aligned}\mathbb{E}[\log S(\mathrm{x}_i|\mathrm{y}^*_{jk},\Sigma,\nu)] \sim\\ -\frac{D}{2}\log(2\pi)-\frac{1}{2}\log|\Sigma|+\frac{D}{2}\mathbb{E}[\log u_{ijk}]\\ -\frac{1}{2}\mathbb{E}[u_{ijk}](\mathrm{x}_i-\mathrm{y}^*_{jk})^T\Sigma^{-1}(\mathrm{x}_i-\mathrm{y}^*_{jk})\\ -\log\Gamma(\frac{\nu}{2})+\frac{\nu}{2}\log\frac{\nu}{2}\\ +(\frac{\nu}{2}-1)\mathbb{E}[\log u_{ijk}]-\frac{\nu}{2}\mathbb{E}[u_{ijk}]\end{aligned} \tag{A.7}$$

where the values of $\mathbb{E}[u_{ijk}]$ and $\mathbb{E}[\log u_{ijk}]$ are given as

$$\begin{aligned}\mathbb{E}[u_{ijk}] &= u^{(\mathrm{new})}_{ijk}\\ &= \frac{\nu^{(\mathrm{new})}+D}{\nu^{(\mathrm{new})}+(\mathrm{x}_i-\mathrm{y}^*_{jk})^T\Sigma^{-1}(\mathrm{x}_i-\mathrm{y}^*_{jk})}\\ \mathbb{E}[\log u_{ijk}] &= \log u^{(\mathrm{new})}_{ijk}\\ &\quad -\log\left(\frac{(\nu^{(\mathrm{new})}+D)}{2}\right)+\psi\left(\frac{(\nu^{(\mathrm{new})}+D)}{2}\right)\end{aligned} \tag{A.8}$$

where $\psi(\cdot)$ is digamma function. From (A.8), omitting terms that are independent of Λ, the objective function $\Im(\Lambda,\Lambda^{(\mathrm{old})})$ in (A.1) is given by

$$\begin{aligned}&\Im(\Lambda,\Lambda^{(\mathrm{old})})\\ &=\sum_{i=1}^{N}(1-z_i)\log w+\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}s_{ijk}\log\upsilon_k\\ &\quad-\frac{D\log\sigma^2}{2}\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}s_{ijk}+\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}s_{ijk}\\ &\quad\times\left(-\log\Gamma\left(\frac{\nu}{2}\right)+\frac{\nu}{2}\log\left(\frac{\nu}{2}\right)-\frac{\nu}{2}u^{(\mathrm{new})}_{ijk}\right.\\ &\quad+\left(\frac{D}{2}+\frac{\nu}{2}-1\right)\left(\log u^{(\mathrm{new})}_{ijk}\right.\\ &\quad\left.\left.-\log\left(\frac{(\nu^{(\mathrm{new})}+D)}{2}\right)+\psi\left(\frac{(\nu^{(\mathrm{new})}+D)}{2}\right)\right)\right)-\frac{1}{2\sigma^2}\\ &\quad\times\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}s_{ijk}u^{(\mathrm{new})}_{ijk}||\mathrm{x}_i-\sum_{l=1}^{M}\varphi_{lk}\mathcal{K}_k(\mathrm{y}_j,\mathrm{y}_l)\\ &\quad-\mathrm{y}_j||^2+\sum_{i=1}^{N}z_i\log(1-w)-\frac{\lambda}{2}\sum_{k=1}^{K}\mathrm{tr}(\Psi_k^T\mathcal{K}_k\Psi_k).\end{aligned} \tag{A.9}$$

From (A.9), the solution of $\partial\Im(\Lambda,\Lambda^{(\mathrm{old})})/\partial\sigma^2=0$, $\partial Q(\Lambda,\Lambda^{(\mathrm{old})})/\partial w=0$, and $\partial\Im(\Lambda,\Lambda^{(\mathrm{old})})/\partial\upsilon_k=0$ [with $\sum_{k=1}^{K}\upsilon_k=1$] yield the estimates of σ^2, w, and υ_k at the new iteration step

$$\begin{aligned}\sigma^2&=\frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}s_{ijk}u^{(\mathrm{new})}_{ijk}||\mathrm{x}_i-\mathrm{y}^*_{jk}||^2}{D\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}s_{ijk}}\\ w&=\frac{1}{N}\sum_{i=1}^{N}(1-z_i);\ \upsilon_k=\frac{\sum_{i=1}^{N}\sum_{j=1}^{M}s_{ijk}}{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{c=1}^{K}s_{ijc}}\end{aligned} \tag{A.10}$$

where $\partial\Im(\Lambda,\Lambda^{(\mathrm{old})})/\partial\Psi_k=0$, we get Ψ_k by

$$\Psi_k=(\mathrm{diag}(\Upsilon_k^T\mathbf{1})\mathcal{K}_k+\lambda\sigma^2\mathbf{I})^{-1}(\Upsilon_k^T\mathrm{x}-\mathrm{diag}(\Upsilon_k^T\mathbf{1})\mathrm{y}). \tag{A.11}$$

Notations used in (A.11) are presented in Table VIII.

TABLE VIII
NOTATIONS USED IN THE PROPOSED METHOD

$\mathrm{x} \in \mathbb{R}^{N \times D}$, $\mathrm{x} = \{x_{id}\}$	The data points (target points)
$\mathrm{y} \in \mathbb{R}^{M \times D}$, $\mathrm{y} = \{y_{jd}\}$	The centroids (model points)
N	The number of points in point set x
M	The number of points in point set y
D	The dimension of the point sets
i = (1,2,...,N)	Index over the observations (target points)
j = (1,2,...,M)	Index over the observations (model points)
d = (1,2,...,D)	Index over the dimensions
$\mathrm{diag}(\cdot)$	The diagonal matrix
$\mathbf{1}$	The column vector of all ones
$\Upsilon_k \in \mathbb{R}^{N \times M}$ $\Upsilon_k = \{s_{ijk} u_{ijk}^{(\mathrm{new})}\}$	The parameter

The estimates of the degrees of freedom ν is given by the solution of the equation

$$-\psi\left(\frac{\nu}{2}\right) + \log\left(\frac{\nu}{2}\right) + \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K} s_{ijk}(\log u_{ijk}^{(\mathrm{new})} - u_{ijk}^{(\mathrm{new})})}{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K} s_{ijk}} + \psi\left(\frac{(\nu^{(\mathrm{new})} + D)}{2}\right) - \log\left(\frac{(\nu^{(\mathrm{new})} + D)}{2}\right) + 1 = 0. \quad \text{(A.12)}$$

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers and the associate editor for their insightful comments that significantly improved the quality of this paper.

REFERENCES

[1] A. Rasoulian, R. Rohling, and P. Abolmaesumi, "Group-wise registration of point sets for statistical shape models," *IEEE Trans. Med. Imag.*, vol. 31, no. 11, pp. 2025–2034, Nov. 2012.
[2] S. Dinggang and C. Davatzikos, "HAMMER: Hierarchical attribute matching mechanism for elastic registration," *IEEE Trans. Med. Imag.*, vol. 21, no. 11, pp. 1421–1439, Nov. 2002.
[3] S. Dinggang and C. Davatzikos, "S-HAMMER: Hierarchical attribute-guided, symmetric diffeomorphic registration for MR brain images," *Hum. Brain Mapp.*, vol. 35, no. 3, pp. 1044–1060, 2014.
[4] M. Ye, X. Wang, R. Yang, L. Ren, and M. Pollefeys, "Accurate 3d pose estimation from a single depth image," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 731–738.
[5] A. Weiss, D. Hirshberg, and M. J. Black, "Home 3D body scans from noisy image and range data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1951–1958.
[6] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 6, pp. 567–585, Jun. 1989.
[7] K. Rohr *et al.*, "Landmark-based elastic registration using approximating thin-plate splines," *IEEE Trans. Med. Imag.*, vol. 20, no. 6, pp. 526–534, Jun. 2001.
[8] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
[9] Z. Zhang, "Iterative point matching for registration of freeform curves and surfaces," *Int. J. Comput. Vis.*, vol. 13, no. 2, pp. 119–152, 1994.
[10] S. Granger and X. Pennec, "Multi-scale EM-ICP: A fast and robust approach for surface registration," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 418–432.
[11] C. L. Tsai, C. Y. Li, G. Yang, and K. S. Lin, "The edge-driven dual-bootstrap iterative closest point algorithm for registration of multimodal fluorescein angiogram sequence," *IEEE Trans. Med. Imag.*, vol. 29, no. 3, pp. 636–649, Mar. 2010.
[12] J. Hermans, D. Smeets, D. Vandermeulen, and P. Suetens, "Robust point set registration using EM-ICP with information-theoretically optimal outlier handling," in *Proc. IEEE Comput. Vis. Pattern Recognit.*, 2011, pp. 2465–2472.
[13] C. V. Stewart, T. C. Ling, and B. Roysam, "The dual-bootstrap iterative closest point algorithm with application to retinal image registration," *IEEE Trans. Med. Imag.*, vol. 22, no. 11, pp. 1379–1394, Nov. 2003.
[14] C. L. Tsai, C. Y. Li, G. Yang, and K. S. Lin, "The edge-driven dual-bootstrap iterative closest point algorithm for registration of multimodal fluorescein angiogram sequence," *IEEE Trans. Med. Imag.*, vol. 29, no. 3, pp. 636–649, Mar. 2010.
[15] Y. Zheng and D. Doermann, "Robust point matching for nonrigid shapes by preserving local neighborhood structures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 643–649, Apr. 2006.
[16] J. Ma, J. Zhao, J. Tian, Z. Tu, and A. L. Yuille, "Robust esitimation of nonrigid transformation for point set registration," in *Proc. IEEE Comput. Vis. Pattern Recognit.*, 2013, pp. 2147–2154.
[17] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Comput. Vis. Image Understand.*, vol. 89, no. 2–3, pp. 114–141, 2003.
[18] J. F. Krucker, G. L. LeCarpentier, J. B. Fowlkes, and P. L. Carson, "Rapid elastic image registration for 3-D ultrasound," *IEEE Trans. Med. Imag.*, vol. 21, no. 11, pp. 1384–1394, Nov. 2002.
[19] A. Myronenko and X. B. Song, "Point-set registration: Coherent point drift," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 12, pp. 2262–2275, Dec. 2010.
[20] A. Myronenko, X. Song, and M. A. Carreira-Perpinan, "Nonrigid point set registration: Coherent point drift," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 2262–2275.
[21] B. Jian and B. Vemuri, "Robust point set registration using Gaussian mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1633–1645, Aug. 2011.
[22] B. Jian and B. Vemuri, "A robust algorithm for point set registration using mixture of Gaussians," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, pp. 1246–1251.
[23] N. Baka *et al.*, "Oriented Gaussian mixture models for nonrigid 2D/3D coronary artery registration," *IEEE Trans. Med. Imag.*, vol. 33, no. 5, pp. 1023–1034, 2014.
[24] A. Myronenko and X. Song, "Intensity-based image registration by minimizing residual complexity," *IEEE Trans. Med. Imag.*, vol. 29, no. 11, pp. 1882–1891, Nov. 2010.
[25] W. Tao and K. Sun, "Asymmetrical Gauss mixture models for point sets matching," in *Proc. IEEE Comput. Vis. Pattern Recognit.*, 2014, pp. 1598–1605.
[26] C. M. Bishop, *Pattern Recognition and Machine Learning*. Berlin, Germany: Springer, 2006.
[27] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, 2004.
[28] P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via EM algorithm," *J. R. Stat. Soc.*, vol. 39, no. 1, pp. 1–38, 1977.
[29] M. N. Thanh and Q. M. J. Wu, "A nonsymmetric mixture model for unsupervised image segmentation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 43, no. 2, pp. 751–765, Apr. 2013.
[30] D. Peel and G. McLachlan, "Robust mixture modeling using the T distribution," *Stat. Comput.*, vol. 10, no. 4, pp. 339–348, 2000.
[31] S. Shoham, "Robust clustering by deterministic agglomeration EM of mixtures of multivariate t-distributions," *Pattern Recognit.*, vol. 35, no. 5, pp. 1127–1142, 2002.
[32] S. P. Chatzis, D. Kosmopoulos, and T. Varvarigou, "Robust sequential data modeling using an outlier tolerant hidden Markov model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 9, pp. 1657–1669, Sep. 2009.
[33] M. N. Thanh and Q. M. J. Wu, "Robust Student's-t mixture model with spatial constraints and its application in medical image segmentation," *IEEE Trans. Med. Imag.*, vol. 21, no. 1, pp. 103–116, Jan. 2012.
[34] Y. Zheng and D. Doermann, "Robust point matching for nonrigid shapes by preserving local neighborhood structures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 643–649, 2006.
[35] R. Castillo *et al.*, "A framework for evaluation of deformable image registration spatial accuracy using large landmark point sets," *Phys. Med. Biol.*, vol. 54, no. 7, pp. 1849–1870, 2009.
[36] E. Castillo, R. Castillo, J. Martinez, M. Shenoy, and T. Guerrero, "Four-dimensional deformable image registration using trajectory modeling," *Phys. Med. Biol.*, vol. 55, no. 1, pp. 305–327, 2010.
[37] R. Castillo *et al.*, "A reference dataset for deformable image registration spatial accuracy evaluation using the COPDgene study archive," *Phys. Med. Biol.*, vol. 58, no. 9, pp. 2861–2877, 2013.
[38] G. A. Ferguson and Y. Takane, *Statistical Analysis in Psychology and Education*. Montreal, QC, Canada: McGraw-Hill, 2005.
[39] A. Mansoor *et al.*, "A generic approach to pathological lung segmentation," *IEEE Trans. Med. Imag.*, vol. 33, no. 12, pp. 2293–2310, Dec. 2014.