Pattern Recognition Letters 32 (2011) 910–918

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

The thin plate spline robust point matching (TPS-RPM) algorithm: A revisit

Jinzhong Yang *

Department of Radiation Physics, The University of Texas MD Anderson Cancer Center, Houston, TX 77030, USA

ARTICLE INFO

Article history:
Received 17 September 2009
Available online 1 February 2011
Communicated by M. Kamel

Keywords:
Robust point matching (RPM)
Thin plate splines (TPS)
Feature extraction
Outliers
Non-rigid registration

ABSTRACT

This paper reviews the TPS-RPM algorithm (Chui and Rangarajan, 2003) for robustly registering two sets of points and demonstrates from a theoretical point of view its inherent limited performance when outliers are present in both point sets simultaneously. A double-sided outlier handling approach is proposed to overcome this limitation with a rigorous mathematical proof as the underlying theoretical support. This double-sided outlier handling approach is proved to be equivalent to the original formulation of the point matching problem. For a practical application, we also extend the TPS-RPM algorithms to non-rigid image registration by registering two sets of sparse features extracted from images. The intensity information of the extracted features are incorporated into feature matching in order to reduce the impact from outliers. Our experiments demonstrate the double-sided outlier handling approach and the efficiency of intensity information in assisting outlier detection.

1. Introduction

Surface matching or image registration problems frequently arise in the medical imaging and computer vision domains (Zitová and Flusser, 2003; Maintz and Viergever, 1998). The registration is a process to establish point correspondence between two surfaces or images. Both surface matching and image registration can boil down to searching for an optimal transformation as well as the correspondence between two sets of points. Typically, in the image registration case, the points could be pixel locations of the images, or some sparse features extracted from the images. In general, the resulting point matching problem is difficult since both the correspondence and transformation are unknown. Furthermore, various factors such as the noise, outliers, and deformations can make the problem even more difficult. In literature, researchers approached this problem normally based on an iterative estimation framework, since optimal solution of the correspondence (transformation) is much easier to produce if the transformation (correspondence) is known. The well-known iterated closest point (ICP) algorithm (Besl and McKay, 1992) is one example to solve for an underlying rigid transform between two point sets. Later developed methods extended the fundamental idea of ICP to address non-rigid mappings of two point sets (Cross and Hancock, 1998; Rohr et al., 2001; Belongie et al., 2002; Chui and Rangarajan, 2003; Zheng and Doermann, 2006; Taron et al., 2009; Gerogiannis et al., 2009; Boughorbel et al., 2010). Some of these methods even possess the capability to detect outliers automatically (Chui and Rangarajan, 2003; Zheng and Doermann, 2006). The outliers here are specifically referred to those points in one point set that have no correspondence in the other point set. These outliers should be identified and rejected during the matching process. In this paper, we particularly focus on solving the point matching problem when a considerable amount of outliers exist simultaneously in both point sets.

The TPS-RPM algorithm (Chui and Rangarajan, 2003) was originally developed to solve the point matching problem in the presence of outliers. This method takes advantage of the softassign technique (Rangarajan et al., 1997) and the deterministic annealing technique (Yuille and Kosowsky, 1994) for robust point matching and it is able to handle outliers existing in the fixed point set (the point set is fixed during matching). However, the inherent structure of TPS-RPM algorithm does not efficiently handle outliers that simultaneously exist in both point sets. The Generalized Robust Point Matching (G-RPM) algorithm (Lin et al., 2003) derived from the TPS-RPM algorithm presented an alternative to solve the problem of outliers in both point sets by incorporating the curvature information. Although the G-RPM algorithm offered a double-sided-outlier handler, it did not provide a rigorous proof to show the equivalence between the original formulation of point matching problem and the energy function of the double-sided-outlier handler. In this paper we will review the TPS-RPM algorithm and demonstrate from a theoretical point of view its inherent limited performance when outliers are present in both point sets simultaneously. We will also present a double-sided outlier handling approach to overcome this limitation with a rigorous mathematical proof as the underlying theoretical support. This double-sided outlier handling approach is proved to be equivalent to the original formulation of the point matching problem. For a practical application, we also extend the TPS-RPM algorithm to non-rigid image registration by registering two sets of sparse

* Tel.: +1 713 792 2814; fax: +1 713 563 2479.
E-mail addresses: jyang4@mdanderson.org, jinzhong.yang@ieee.org

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.01.015

features extracted from two images. The intensity information of the extracted features are incorporated to regularize the overall energy function of TPS-RPM algorithm in order to reduce the impact from outliers, which are resulted from automatic feature extraction, the common case in most real applications. Our experiments demonstrate the double-sided outlier handling approach and the efficiency of intensity information in assisting outlier detection.

The remainder of this paper is organized as follows. A review of the TPS-RPM algorithm and the analysis of its outlier handling function are given in Section 2. In Section 3 we provide a mathematical proof to demonstrate a double-sided outlier handling approach. Section 4 extends the TPS-RPM algorithm to image registration by incorporating intensity information. Experimental results for both point matching and image registration are provided in Section 5. Finally, we conclude our work in Section 6.

2. Review of the TPS-RPM algorithm

Given two point sets, the moving point set $X = \{\boldsymbol{x}_i : i = 1,2,\ldots,L\}$ and the fixed point set $Y = \{\boldsymbol{y}_j : j = 1,2,\ldots,N\}$ ($X, Y \in \Re^2$ for the sake of simplicity in presentation), the TPS-RPM algorithm detects the correspondence between X and Y and match these two point sets according to a smooth non-rigid transformation. To describe the algorithm, we first introduce the notation for the correspondence matrix and the spatial transformation. A binary *correspondence matrix* $\boldsymbol{P}$ with dimension $(L+1)\times(N+1)$ is defined to characterize the correspondence between X and Y,

$$\boldsymbol{P} = \left(\begin{array}{ccc|c} p_{11} & \cdots & p_{1N} & p_{1,N+1} \\ \vdots & \ddots & \vdots & \vdots \\ p_{L1} & \cdots & p_{LN} & p_{L,N+1} \\ \hline p_{L+1,1} & \cdots & p_{L+1,N} & 0 \end{array}\right). \tag{1}$$

The matrix $\boldsymbol{P}$ consists of two parts. The $L\times N$ inner submatrix defines the correspondence between X and Y. If $\boldsymbol{x}_i$ corresponds to $\boldsymbol{y}_j$, then $p_{ij} = 1$, otherwise $p_{ij} = 0$. The $(N+1)$th column and the $(L+1)$th row define the outliers in X and Y, respectively. If $\boldsymbol{x}_i$ (or $\boldsymbol{y}_j$) is an outlier, then $p_{i,N+1} = 1$ (or $p_{L+1,j} = 1$). $\boldsymbol{P}$ satisfies the row and column normalization conditions, $\sum_{j=1}^{N+1} p_{ij} = 1$, for $i = 1,2,\ldots,L$, and $\sum_{i=1}^{L+1} p_{ij} = 1$, for $j = 1,2,\ldots,N$. Let f denote a non-rigid *spatial transformation* that is used to characterize the non-rigid transform between X and Y. Under the mapping f, the point set X is transformed to $X' = \{\boldsymbol{x}'_i = f(\boldsymbol{x}_i) : i = 1,2,\ldots,L\}$.

2.1. Objective function

The TPS-RPM algorithm minimizes the following energy function for point matching,

$$\begin{aligned} &[\hat{\boldsymbol{P}},\hat{f}] = \arg\min_{\boldsymbol{P},f} E(\boldsymbol{P},f), \\ &E(\boldsymbol{P},f) = \sum_{j=1}^{N}\sum_{i=1}^{L} p_{ij}\|\boldsymbol{y}_j - f(\boldsymbol{x}_i)\|^2 + \lambda\|\mathcal{L}f\|^2 - \zeta\sum_{j=1}^{N}\sum_{i=1}^{L} p_{ij}, \end{aligned} \tag{2}$$

subject to the following constraints,

$$\begin{cases} p_{ij} \in \{0,1\}, & \text{for } i = 1,2,\ldots,L+1;\ j = 1,2,\ldots,N+1. \\ \sum_{j=1}^{N+1} p_{ij} = 1, & \text{for } i = 1,2,\ldots,L. \\ \sum_{i=1}^{L+1} p_{ij} = 1, & \text{for } j = 1,2,\ldots,N. \end{cases} \tag{3}$$

Here $\mathcal{L}$ in (2) is an operator denoting the smoothness regularization. Specifically, the thin-plate splines (TPS) are applied here,

$$\|\mathcal{L}f\|^2 = \int\int \left[\left(\frac{\partial^2 f}{\partial u^2}\right)^2 + 2\left(\frac{\partial^2 f}{\partial u\partial v}\right)^2 + \left(\frac{\partial^2 f}{\partial v^2}\right)^2\right] du\,dv, \tag{4}$$

where u and v represent the two dimensions of the points. The last term in (2) is designed to prevent excessive outlier rejection. λ and ζ are the weighting parameters to balance these terms.

Solving objective function (2) results into two coupled optimization problems: one is the correspondence problem, which is cast as a discrete linear assignment problem (Papadimitriou and Steiglitz, 1998), and the other is the transformation problem, which boils down to a least-square continuous optimization problem. The TPS-RPM algorithm adopts the softassign and deterministic annealing techniques to convert the discrete correspondence problem to a continuous one and thus achieves a robust point matching. The basic idea of softassign (Rangarajan et al., 1997) is to relax the binary correspondence matrix $\boldsymbol{P}$ to take values from interval $[0,1]$, while the row and column normalization constraints are enforced via the iterated row and column normalization method (Sinkhorn, 1964). The deterministic annealing technique (Yuille and Kosowsky, 1994) is introduced to control the behavior of the fuzzy correspondence. This technique is a heuristic continuation method that attempts to find the global optimum of the energy function at high temperature and tracks it as the temperature decreases. An entropy term, $T\sum_{j=1}^{N}\sum_{i=1}^{L} p_{ij}\log p_{ij}$, is added to the energy function (2) for this purpose. Here T is called the temperature parameter, which decreases gradually during optimization. This entropy term enables the fuzzy correspondence matrix to improve gradually and continuously during the optimization process instead of jumping around in the space of binary permutation matrices and outliers. When T reaches zero, $\boldsymbol{P}$ becomes binary and the outliers are identified naturally from the matrix (1). The objective function with a fuzzy correspondence matrix (to simplify notation, we use $\boldsymbol{P}$ as well) is formulated as

$$\begin{aligned} E(\boldsymbol{P},f) = &\sum_{j=1}^{N}\sum_{i=1}^{L} p_{ij}\|\boldsymbol{y}_j - f(\boldsymbol{x}_i)\|^2 + \lambda\|\mathcal{L}f\|^2 + T\sum_{j=1}^{N}\sum_{i=1}^{L} p_{ij}\log p_{ij} \\ &- \zeta\sum_{j=1}^{N}\sum_{i=1}^{L} p_{ij}, \end{aligned} \tag{5}$$

subject to

$$\begin{cases} 0 \leqslant p_{ij} \leqslant 1, & \text{for } i = 1,2,\ldots,L+1;\ j = 1,2,\ldots,N+1. \\ \sum_{j=1}^{N+1} p_{ij} = 1, & \text{for } i = 1,2,\ldots,L. \\ \sum_{i=1}^{L+1} p_{ij} = 1, & \text{for } j = 1,2,\ldots,N. \end{cases} \tag{6}$$

2.2. The TPS-RPM algorithm

The TPS-RPM algorithm solves the optimization problem in (5) and (6) as follow, which is similar to the EM algorithm involving a dual update process embedded within an annealing scheme. The two-step update is briefly summarized here.

- **Step 1:** Update the correspondence $\boldsymbol{P}$ by fixing the transformation f. For correspondence points $i = 1,2,\ldots,L$ and $j = 1,2,\ldots,N$,

$$p_{ij} = \frac{1}{T}\exp\left[\frac{\zeta}{T} - \frac{(\boldsymbol{y}_j - f(\boldsymbol{x}_i))^T(\boldsymbol{y}_j - f(\boldsymbol{x}_i))}{T}\right] \tag{7}$$

and for outliers possibilities in X, $i = 1,2,\ldots,L$ and $j = N+1$,

$$p_{i,N+1} = \frac{1}{T_0}\exp\left[-\frac{(\boldsymbol{y}_{N+1} - f(\boldsymbol{x}_i))^T(\boldsymbol{y}_{N+1} - f(\boldsymbol{x}_i))}{T_0}\right] \tag{8}$$

and for outliers possibilities in Y, $i = L+1$ and $j = 1,2,\ldots,N$,

$$p_{L+1,j} = \frac{1}{T_0}\exp\left[-\frac{(\boldsymbol{y}_j - \boldsymbol{x}_{L+1})^T(\boldsymbol{y}_j - \boldsymbol{x}_{L+1})}{T_0}\right], \tag{9}$$

where $\boldsymbol{x}_{L+1}$ and $\boldsymbol{y}_{N+1}$ are outlier cluster centers and T_0 is a very large variance. Iterated row and column normalization algorithm is then applied to the resulted $\boldsymbol{P}$ in (7)–(9) to satisfy the constraints (6),

$$p_{ij} = \frac{p_{ij}}{\sum_{l=1}^{N+1} p_{il}}, \quad i = 1,2,\ldots,L;\ j = 1,2,\ldots,N+1, \tag{10}$$

$$p_{ij} = \frac{p_{ij}}{\sum_{l=1}^{L+1} p_{lj}}, \quad i = 1,2,\ldots,L+1;\ j = 1,2,\ldots,N. \tag{11}$$

- **Step 2:** Update the transformation f by fixing the correspondence $\boldsymbol{P}$. The TPS-RPM algorithm simplify this step by minimizing the following standard TPS bending energy function, a simplified form of Eq. (5):

$$E^0_{TPS}(f) = \sum_{i=1}^{L} \|\boldsymbol{z}_i - f(\boldsymbol{x}_i)\|^2 + \lambda\|\mathcal{L}f\|^2, \tag{12}$$

where

$$\boldsymbol{z}_i = \sum_{j=1}^{N} p_{ij}\boldsymbol{y}_j. \tag{13}$$

Let $\boldsymbol{X}$ and $\boldsymbol{Z}$, each of dimension $L \times 3$, denote the stacked versions of the homogenous point coordinates $\boldsymbol{x}_i$ and $\boldsymbol{z}_i$, respectively, and $\boldsymbol{K}$ the TPS kernel of dimension $L \times L$, the stacked version of vector $\boldsymbol{k}(\boldsymbol{x}_i)$ with each entry $k_{ji} = \boldsymbol{k}_j(\boldsymbol{x}_i) = \|\boldsymbol{x}_j - \boldsymbol{x}_i\|^2\log\|\boldsymbol{x}_j - \boldsymbol{x}_i\|$, for $i = 1,2,\ldots,L$; $j = 1,2,\ldots,L$. The minimization of (12) produces a unique solution for f that is composed of $[\boldsymbol{a},\boldsymbol{w}]$ as

$$f(\boldsymbol{x}_i) = \boldsymbol{x}_i \cdot \boldsymbol{a} + \boldsymbol{k}(\boldsymbol{x}_i)\cdot\boldsymbol{w}, \tag{14}$$

where $\boldsymbol{a}$ is a 3×3 matrix representing the affine transform and $\boldsymbol{w}$ is an $L \times 3$ matrix representing the non-affine warping transform. By applying QR decomposition to $\boldsymbol{X}$,

$$\boldsymbol{X} = (\boldsymbol{Q}_1\boldsymbol{Q}_2)\begin{pmatrix}\boldsymbol{R}\\ \boldsymbol{0}\end{pmatrix}, \tag{15}$$

where $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ are $L \times 3$ and $L \times (L-3)$ orthonormal matrices, respectively, and $\boldsymbol{R}$ is a 3×3 upper triangular matrix, the TPS-RPM algorithm produces the optimal solution $[\hat{\boldsymbol{a}}^0, \hat{\boldsymbol{w}}^0]$ as

$$\hat{\boldsymbol{w}}^0 = \boldsymbol{Q}_2\left(\boldsymbol{Q}_2^T\boldsymbol{K}\boldsymbol{Q}_2 + \lambda\boldsymbol{I}_{(L-3)}\right)^{-1}\boldsymbol{Q}_2^T\boldsymbol{Z}, \tag{16}$$

$$\hat{\boldsymbol{a}}^0 = \boldsymbol{R}^{-1}\boldsymbol{Q}_1^T(\boldsymbol{Z} - \boldsymbol{K}\hat{\boldsymbol{w}}^0), \tag{17}$$

where $\boldsymbol{I}_{(L-3)}$ is an $(L-3)\times(L-3)$ identity matrix.

The readers are referred to Chui and Rangarajan (2003) for more details regarding the TPS-RPM algorithm. The development of this algorithm was also formulated in a probability framework with Gaussian mixture models (Chui and Rangarajan, 2000). In this framework, the data points in the fixed point set are treated as random variables with Gaussian mixture density, with the Gaussian cluster centers determined by the points in the moving point set. The outliers are modeled as an extra Gaussian cluster with a very large variance. This formulation of Gaussian mixture models may be helpful to understand the computation of outlier possibilities in (8) and (9).

2.3. Outlier rejection

As concerning the automatic outlier rejection, two issues should be addressed in an algorithm: the first is to identify outliers automatically from corresponding points; and the second is to ensure no outliers are applied to the computation of transformation. In the TPS-RPM algorithm, Eqs. (8) and (9) serve as the detection of outliers in both point sets. The probability of a point in X being an outlier, $p_{i,N+1}$, and the probability of a point in Y being an outlier, $p_{L+1,j}$, are determined after the iterated normalization in (10) and (11) are applied. However, the awkwardness here is that, the outlier cluster centers $\boldsymbol{x}_{L+1}$ and $\boldsymbol{y}_{N+1}$ are unknown, and it is difficult to determine the large variance T_0. The TPS-RPM algorithm simply sets each outlier cluster center at the center of mass of each point set. Nevertheless, it is not a good choice since the defined outlier cluster may easily include corresponding points. Furthermore, the normalization constraints in (6) imply that the probability of a point being an outlier depends on its correspondence probabilities, that is, it can be inferred from the values of correspondence points, p_{ij}, $i = 1,2,\ldots,L$; $j = 1,2,\ldots,N$. Therefore, the values of $p_{i,N+1}$ and $p_{L+1,j}$ are determined in each iteration after the computation of Eq. (7) followed by the alternating row and column normalization (10) and (11), yet including Eqs. (8) and (9) for normalization may result into an ill-posed problem for the convergence of correspondence matrix $\boldsymbol{P}$.

To determine a correct transformation, it is critical to exclude all outliers from computing for it, specifically, the TPS transform in this case. The TPS-RPM algorithm minimizes Eq. (12) to solve for an optimal TPS transform in each iteration. If $\boldsymbol{y}_j$ is identified as an outlier, which implies $p_{ij} \to 0$ for $i = 1,2,\ldots,L$, from the observation of Eq. (13) we know it will not contribute to $\boldsymbol{z}_i$ in (12) for computing the TPS transformation. However, if $\boldsymbol{x}_i$ is an outlier, there is no appropriate means to exclude its contribution in (12). Since $\boldsymbol{x}_i$ being an outlier implies $p_{ij} \to 0$ for $j = 1,2,\ldots,N$, point $\boldsymbol{z}_i$, the correspondence of $\boldsymbol{x}_i$, is $(0,0)$. Therefore, in each iteration, the TPS-RPM algorithm actually tries to match all outliers in the moving point set to a virtual point in the fixed point set, which is supposed to be the origin of the coordinate system. This issue is actually resulted from the simplified TPS bending energy function (12), which is not equivalent to the originally formulated energy function (5), in computing optimal f with fixed $\boldsymbol{P}$. This simplification implicitly reduces the capability of both-side outlier handling in (5) so that unexpected erroneous result may arise when the outliers present in the moving point set. In next section, we will demonstrate an approach that is directly derived from Eq. (5) for handling outliers in both point sets.

3. Double-sided outlier handling

In last section, we have shown that the TPS-RPM algorithm reduces its capability of handling outliers in both sides since it utilizes the simplified energy function (12) for calculating the TPS transformation. Rather than using this simplified form, we demonstrate that it is possible to directly optimize the objective function (5). This optimization can be simplified according to the following theorem while keeping the property of double-sided outlier handling.

Theorem 3.1. *Minimizing objective function (5) w.r.t. transformation f is equivalent to minimizing the following TPS bending energy function,*

$$E^1_{TPS}(f) = \frac{1}{N}\sum_{i=1}^{L}\frac{\|\bar{\boldsymbol{z}}_i - f(\boldsymbol{x}_i)\|^2}{\delta_i^2} + \lambda\|\mathcal{L}f\|^2, \tag{18}$$

where

$$\bar{\boldsymbol{z}}_i = \frac{\sum_{j=1}^{N} p_{ij}\boldsymbol{y}_j}{\sum_{j=1}^{N} p_{ij}}, \quad i = 1,\ldots,L \tag{19}$$

and

$$\delta_i^2 = \frac{1}{\sum_{j=1}^{N} p_{ij}}, \quad i = 1, \ldots, L. \tag{20}$$

Proof. Notice that the last two terms in (5) do not contain f, therefore, to minimize (5) w.r.t. the transformation f, one can equally minimize the following energy function,

$$E_1(f) = \sum_{j=1}^{N} \sum_{i=1}^{L} p_{ij} \|\boldsymbol{y}_j - f(\boldsymbol{x}_i)\|^2 + \lambda \|\mathcal{L}f\|^2. \tag{21}$$

Let $\boldsymbol{d}_i = (\|\boldsymbol{y}_1 - f(\boldsymbol{x}_i)\|, \ldots, \|\boldsymbol{y}_N - f(\boldsymbol{x}_i)\|)_N^T$ and $\boldsymbol{\Sigma}_i = diag\{p_{i1}, \ldots, p_{iN}\}$, then Eq. (21) can be rewritten as

$$E_1(f) = \sum_{i=1}^{L} \boldsymbol{d}_i^T \boldsymbol{\Sigma}_i \boldsymbol{d}_i + \lambda \|\mathcal{L}f\|^2. \tag{22}$$

As aforementioned, the transformation f is composed of $[\boldsymbol{a}, \boldsymbol{w}]$, and the optimal solution to (22) can be efficiently computed from the following linear equations (Wahba, 1990; Wang, 1998; Rohr et al., 1999),

$$(\tilde{\boldsymbol{K}} + \lambda \tilde{\boldsymbol{W}}^{-1})\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{X}}\boldsymbol{a} = \tilde{\boldsymbol{Y}}, \tag{23a}$$

$$\tilde{\boldsymbol{X}}^{\boldsymbol{T}}\tilde{\boldsymbol{w}} = \boldsymbol{0}, \tag{23b}$$

where $\tilde{\boldsymbol{K}}$ is an $(L \cdot N) \times (L \cdot N)$ matrix, $\tilde{\boldsymbol{K}} = \begin{pmatrix} k_{11}\boldsymbol{I}_N & \cdots & k_{1L}\boldsymbol{I}_N \\ \vdots & \ddots & \vdots \\ k_{L1}\boldsymbol{I}_N & \cdots & k_{LL}\boldsymbol{I}_N \end{pmatrix}$, with $\boldsymbol{I}_N$ the $N \times N$ identity matrix, and $\tilde{\boldsymbol{W}}$ is an $(L \cdot N) \times (L \cdot N)$ diagonal matrix, $\tilde{\boldsymbol{W}} = diag\{\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_L\}$. Let $\boldsymbol{1}_N = (1, 1, \ldots, 1)_N^T$ denote a vector of length N with value 1, and $\boldsymbol{Y}$ the stacked version of homogenous point coordinates $\boldsymbol{y}_j$, $j = 1, \ldots, N$, with dimension $N \times 3$. Then, $\tilde{\boldsymbol{w}}, \tilde{\boldsymbol{X}}$, and $\tilde{\boldsymbol{Y}}$ in (23) can be written as follows: $\tilde{\boldsymbol{w}} = (\boldsymbol{1}_N \cdot \boldsymbol{w}_1, \ldots, \boldsymbol{1}_N \cdot \boldsymbol{w}_L)_L^T$, where $\boldsymbol{w}_i$ is the ith row vector of $\boldsymbol{w}$, $\tilde{\boldsymbol{X}} = (\boldsymbol{1}_N \cdot \boldsymbol{x}_1, \ldots, \boldsymbol{1}_N \cdot \boldsymbol{x}_L)_L^T$, and $\tilde{\boldsymbol{Y}} = (\boldsymbol{Y}^T, \ldots, \boldsymbol{Y}^T)_L^T$ is an $(L \cdot N) \times 3$ matrix which is stacked together of L number of $\boldsymbol{Y}$s.

It is straightforward that Eq. (23b) can be simplified to $\boldsymbol{X}^T\boldsymbol{w} = \boldsymbol{0}$ by removing the redundant equations. Now we check Eq. (23a): there are total $L \times N$ equations and these equations can be separated into L groups with each group N equations. For the ith group, $i = 1, 2, \ldots, L$, these N equations can be written out as

$$\begin{aligned} &\lambda p_{i1}^{-1}\boldsymbol{w}_i + \sum_{l=1}^{L} k_{il}\boldsymbol{w}_l + \boldsymbol{x}_i \cdot \boldsymbol{a} = \boldsymbol{y}_1, \\ &\lambda p_{i2}^{-1}\boldsymbol{w}_i + \sum_{l=1}^{L} k_{il}\boldsymbol{w}_l + \boldsymbol{x}_i \cdot \boldsymbol{a} = \boldsymbol{y}_2, \\ &\vdots \\ &\lambda p_{iN}^{-1}\boldsymbol{w}_i + \sum_{l=1}^{L} k_{il}\boldsymbol{w}_l + \boldsymbol{x}_i \cdot \boldsymbol{a} = \boldsymbol{y}_N. \end{aligned} \tag{24}$$

For the jth equation in (24), $j = 1, 2, \ldots, N$, multiply p_{ij} on the both sides of that equation, and sum all these N equations to produce

$$N\lambda\boldsymbol{w}_i + \sum_{j=1}^{N} p_{ij}\left(\sum_{l=1}^{L} k_{il}\boldsymbol{w}_l + \boldsymbol{x}_i \cdot \boldsymbol{a}\right) = \sum_{j=1}^{N} p_{ij}\boldsymbol{y}_j. \tag{25}$$

Now plug (19) and (20) into (25), we obtain

$$N\lambda(\delta_i^2)^{-1}\boldsymbol{w}_i + \sum_{l=1}^{L} k_{il}\boldsymbol{w}_l + \boldsymbol{x}_i \cdot \boldsymbol{a} = \bar{\boldsymbol{z}}_i, \quad i = 1, 2, \ldots, L. \tag{26}$$

Rewrite (26) in the matrix format, and the optimal solution $f = [\boldsymbol{a}, \boldsymbol{w}]$ to the objective function (21) can be computed equivalently from these linear equations,

$$(\boldsymbol{K} + N\lambda\boldsymbol{W}^{-1})\boldsymbol{w} + \boldsymbol{X}\boldsymbol{a} = \overline{\boldsymbol{Z}}, \tag{27a}$$

$$\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{w} = \boldsymbol{0}, \tag{27b}$$

where $\boldsymbol{W} = diag\{\delta_1^2, \delta_2^2, \ldots, \delta_L^2\}$ and $\overline{\boldsymbol{Z}}$ is the stacked version of $\bar{\boldsymbol{z}}_i$. It is easy to verify that the solution to Eq. (27) is also the optimal solution to objective function (18) (Wang, 1998). Therefore, minimizing Eq. (21) is equivalent to minimizing Eq. (18), thus proving the theorem. □

Lemma 3.2. (Double-sided outlier handling) *Objective function (18) implicitly detects outliers in both point sets X and Y and rejects them from computing for an optimal TPS transform.*

Proof. If $\boldsymbol{y}_j$ in fixed point set Y is identified as an outlier, which implies $p_{ij} \to 0$ for $i = 1, 2, \ldots, L$, from Eq. (19) we know it will not contribute to $\bar{\boldsymbol{z}}_i$ in (18) for computing an optimal transformation. If $\boldsymbol{x}_i$ in moving point set X is identified as an outlier, which implies $p_{ij} \to 0$ for $j = 1, 2, \ldots, N$, from Eq. (20) we know $1/\delta_i^2 \to 0$, which is equivalent to removing the contribution of $\boldsymbol{x}_i$ from the computation in (18). Therefore, objective function (18) implements double-sided outlier handling. □

Lin et al. (2003) presented a similar objective function as (18) for double-sided outlier handling, though it does not rigorously prove the equivalence to the original formulation of point matching problem. We demonstrated here with a rigorous mathematical proof the feasibility of this double-sided outlier handler, thus providing the theoretical foundation to the double-sided outlier handling in the TPS-RPM algorithm.

Lemma 3.3. *The optimal solution* $\hat{f}^1 = [\hat{\boldsymbol{a}}^1, \hat{\boldsymbol{w}}^1]$ *to (18) is*

$$\hat{\boldsymbol{w}}^1 = \boldsymbol{Q}_2\left(\boldsymbol{Q}_2^T(\boldsymbol{K} + N\lambda\boldsymbol{W})\boldsymbol{Q}_2\right)^{-1}\boldsymbol{Q}_2^T\overline{\boldsymbol{Z}}, \tag{28}$$

$$\hat{\boldsymbol{a}}^1 = \boldsymbol{R}^{-1}\boldsymbol{Q}_1^T(\overline{\boldsymbol{Z}} - (\boldsymbol{K} + N\lambda\boldsymbol{W})\hat{\boldsymbol{w}}^1). \tag{29}$$

Proof. The result is straightforward by applying the QR decomposition to $\boldsymbol{X}$ (see Eq. (15)) and solving the linear equations in (27) (Wahba, 1990). □

4. Extend to image registration

Image registration is a process to establish point correspondence between two images of the same scene. As previously stated in Section 1, point sets X and Y could also be the sparse geometric features extracted from two images that we would like to register. Therefore, the TPS-RPM algorithm can be directly applied to image registration. To formulate the matching problem, X and Y are two sets of features extracted from the reference image and the alignment image, respectively, and they should cover most interesting parts in each input image. The resulting transformation f from the TPS-RPM algorithm characterizes the underlying mapping from the reference image to the alignment image, and it can be used to resample the alignment image to obtain a registered one. The registration process is considered proper if f is able to transform each voxel of the reference image to its corresponding location in the alignment image. In general, the features X and Y are extracted automatically and contain a large amount of outliers in both sides. Therefore, the detection and rejection of outliers are imperative during the registration process. In the TPS-RPM algorithm, the matching criterion is solely based on the Euclidean distance of points. Even if equipped with the double-sided outlier handler described in Section 3, the TPS-RPM algorithm is still not efficient enough to discriminate all those outliers since it may identify some pairs of outliers as correspondence points if they are spatially close. Therefore, additional information is necessary to boost the registration.

Intensity value associated with each point in the image provides rich information that is helpful to identify matching correspondence during registration process if it is properly presented. The salient region feature proposed by Kadir and Brady (2001) uses a circular region of intensity values to describe the uniqueness and saliency of the centroid point of this region and has long been

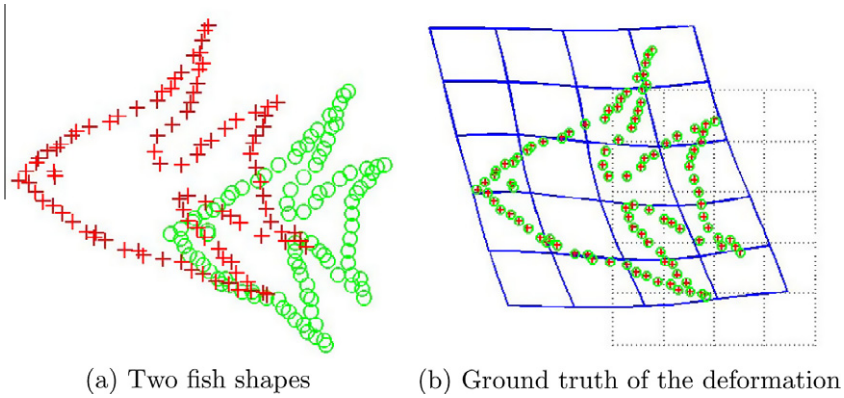

Fig. 1. Two fish shapes and the ground truth of the deformation grids between them.

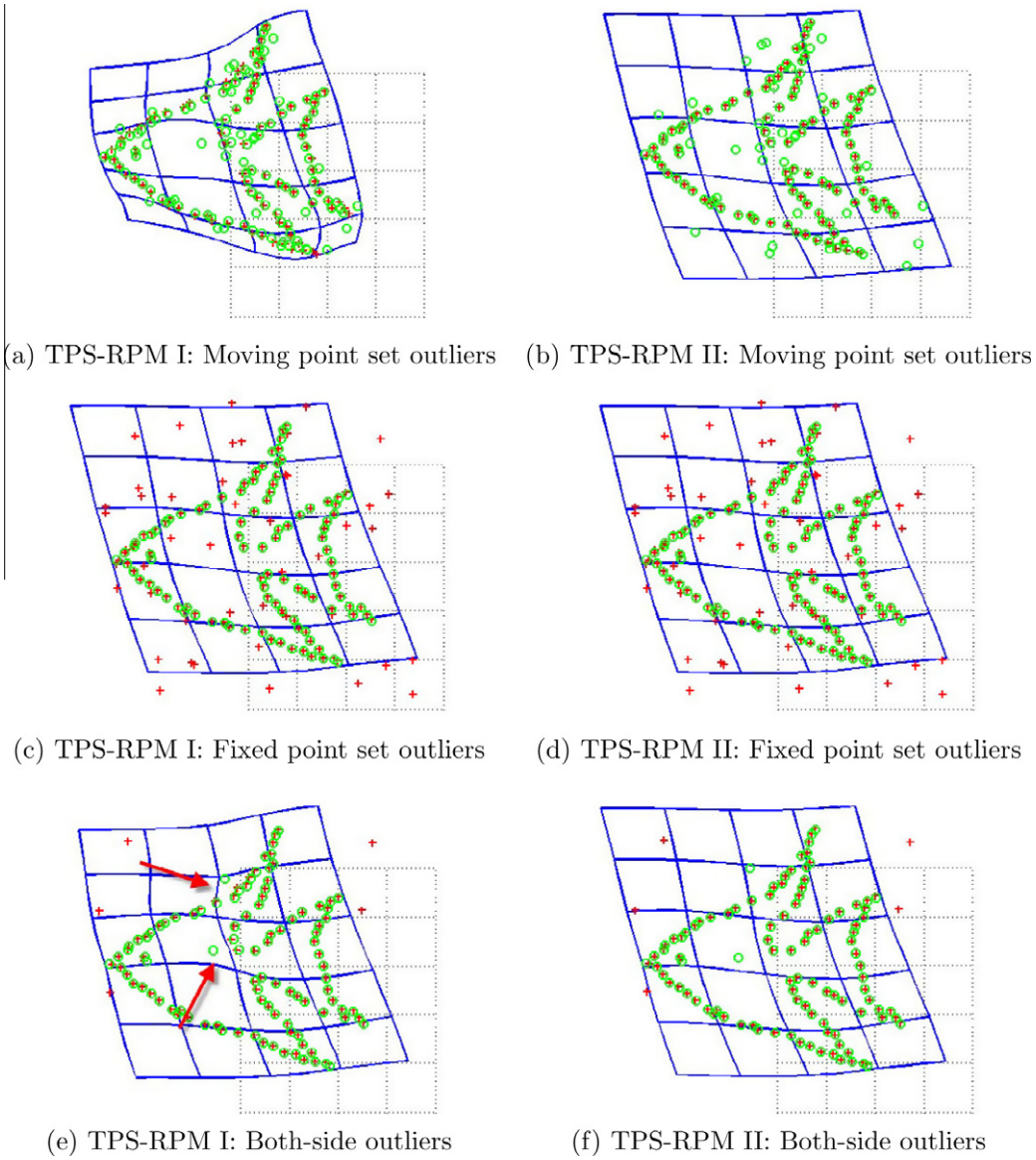

Fig. 2. Comparison of the original TPS-RPM algorithm (TPS-RPM I) and the revised TPS-RPM algorithm (TPS-RPM II) in outlier handling.

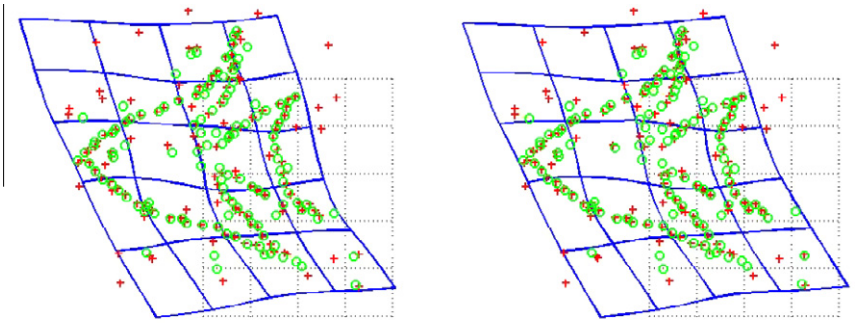

Fig. 3. Comparison of the original TPS-RPM algorithm (TPS-RPM I) and the revised TPS-RPM algorithm (TPS-RPM II) in terms of outlier handling when a considerable amount of outliers present in both the moving and fixed point sets.

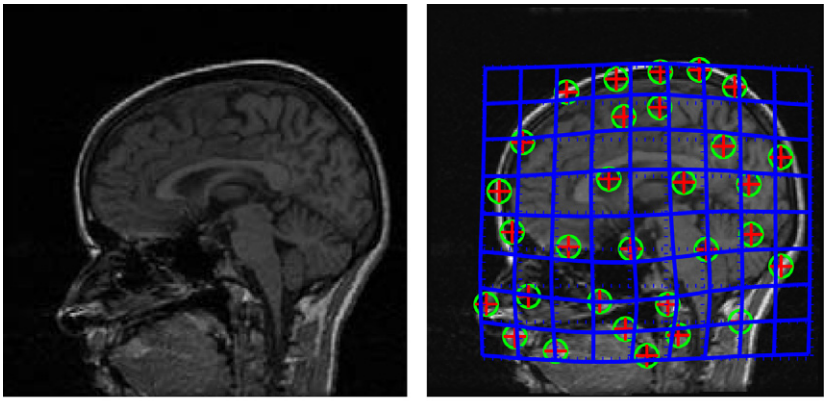

Fig. 4. The brain image and the simulated deformed image with the ground truth of deformation shown in grids.

attractive in the image registration community (Huang et al., 2004; Wu et al., 2005; Yang et al., 2006). Yang et al. (2006) integrated this feature into the framework of the TPS-RPM algorithm to achieve robust non-rigid image registration. Specifically, let $\mathcal{R}(\boldsymbol{x})$ denote the salient region centered at point $\boldsymbol{x}$, then a new energy term, $E_d(\boldsymbol{P},f)$, which is defined in the sense of salient feature dissimilarity, has been added to the overall objective function (5),

$$E_d(\boldsymbol{P},f) = \sigma \sum_{j=1}^{N}\sum_{i=1}^{L} p_{ij}\left\{1 - \Psi\left[\mathcal{R}(\boldsymbol{y}_j), \mathcal{R}(f(\boldsymbol{x}_i))\right]\right\}^2, \tag{30}$$

where σ is the prior weight for the importance of $E_d(\boldsymbol{P},f)$, and $\Psi[\cdot,\cdot]$ measures the similarity of two salient regions, with the range from 0 to 1 where 1 indicates two most similar salient regions. $E_d(\boldsymbol{P},f)$ controls how the salient region features are employed in finding correspondence. In order to adjust the impact of $E_d(\boldsymbol{P},f)$ to be similar with that of Euclidean distance-based energy term in (5), a quadratic term is used in (30). With this new energy term, the correspondence detection function in (7) is updated to be

$$p_{ij} = \frac{1}{T}\exp\left[\frac{\zeta}{T} - \frac{(\boldsymbol{y}_j - f(\boldsymbol{x}_i))^T(\boldsymbol{y}_j - f(\boldsymbol{x}_i))}{T} - \frac{\sigma\left\{1 - \Psi\left[\mathcal{R}(\boldsymbol{y}_j), \mathcal{R}(f(\boldsymbol{x}_i))\right]\right\}^2}{T}\right]. \tag{31}$$

With the additional information from salient feature similarity, Eq. (31) is more discriminable than Eq. (7) in detecting outliers. Practically, the similarity function $\Psi[\cdot,\cdot]$ can be implemented efficiently with the normalized local cross-correlation coefficient (LCC) (Yang et al., 2006), or the entropy correlation coefficient (ECC) (Huang et al., 2004). For LCC, the time for computing the salient feature similarity is $O(N^2n)$, with N points in each point set and the average size of a salient region of n pixels.

5. Experimental results

In this section, we will demonstrate the theoretical analysis of the outlier handling in the TPS-RPM algorithm with simulated examples and promote the revised TPS-RPM algorithm with the double-sided outlier handler. We will also illustrate the efficacy of the intensity information in assisting the outlier detection for image registration.

The first example is a shape registration which reveals the capability of double-sided outlier handling in the revised TPS-RPM algorithm. The fish shapes shown in Fig. 1(a) are used for example. We will register the shape in green circles[1] (moving point set) to the shape in red pluses (fixed point set). The ground truth of the deformation is shown in Fig. 1(b). Fig. 2 shows the registration results from the original TPS-RPM algorithm (TPS-RPM I) and the revised TPS-RPM algorithm (TPS-RPM II) with double-sided outlier handler. In the first test, we randomly add some outliers to the moving point set and perform the registration. Fig. 2(a) shows the result from the TPS-RPM I algorithm, and Fig. 2(b) shows the

[1] For interpretation of color in Figs. 1–6, the reader is referred to the web version of this article.

result from the TPS-RPM II algorithm. It is obvious that the TPS-RPM I algorithm is not able to detect and reject the outliers in the moving point set from matching. All outliers are pulled toward their closest points in the fixed point set. However, the TPS-RPM II algorithm can identify all those outliers correctly. In the second test, we randomly add some outliers to the fixed point set and the registration results are shown in Fig. 2(c) and (d). In this test, both algorithms identify outliers correctly and achieve good registration results. In the third test, we added a small number of outliers to the both point sets. In observation of the registration result from TPS-RPM I in Fig. 2(e), the outliers in the moving point sets have an obvious impact on the registration result by competing with other corresponding points during the matching process. However, the TPS-RPM II is able to identify outliers in both point sets and obtain the correct deformation (Fig. 2(f)). To this point, we have demonstrated the double-sided outlier handling of the TPS-RPM II algorithm in this example. Next we look at another test when a considerable amount of outliers present in both point sets simultaneously. Fig. 3(a) and (b) shows the registration results from both TPS-RPM algorithms in this test. Both results are not consistent with the ground truth in Fig. 1(b) since both algorithms identify some pairs of outliers as correspondence points. It reveals that the Euclidean distance only is not able to distinguish those outliers. Additional information, such as the curvature information (Lin et al., 2003), could be included for the purpose of rejecting outliers in order to achieve a good point matching result. The readers

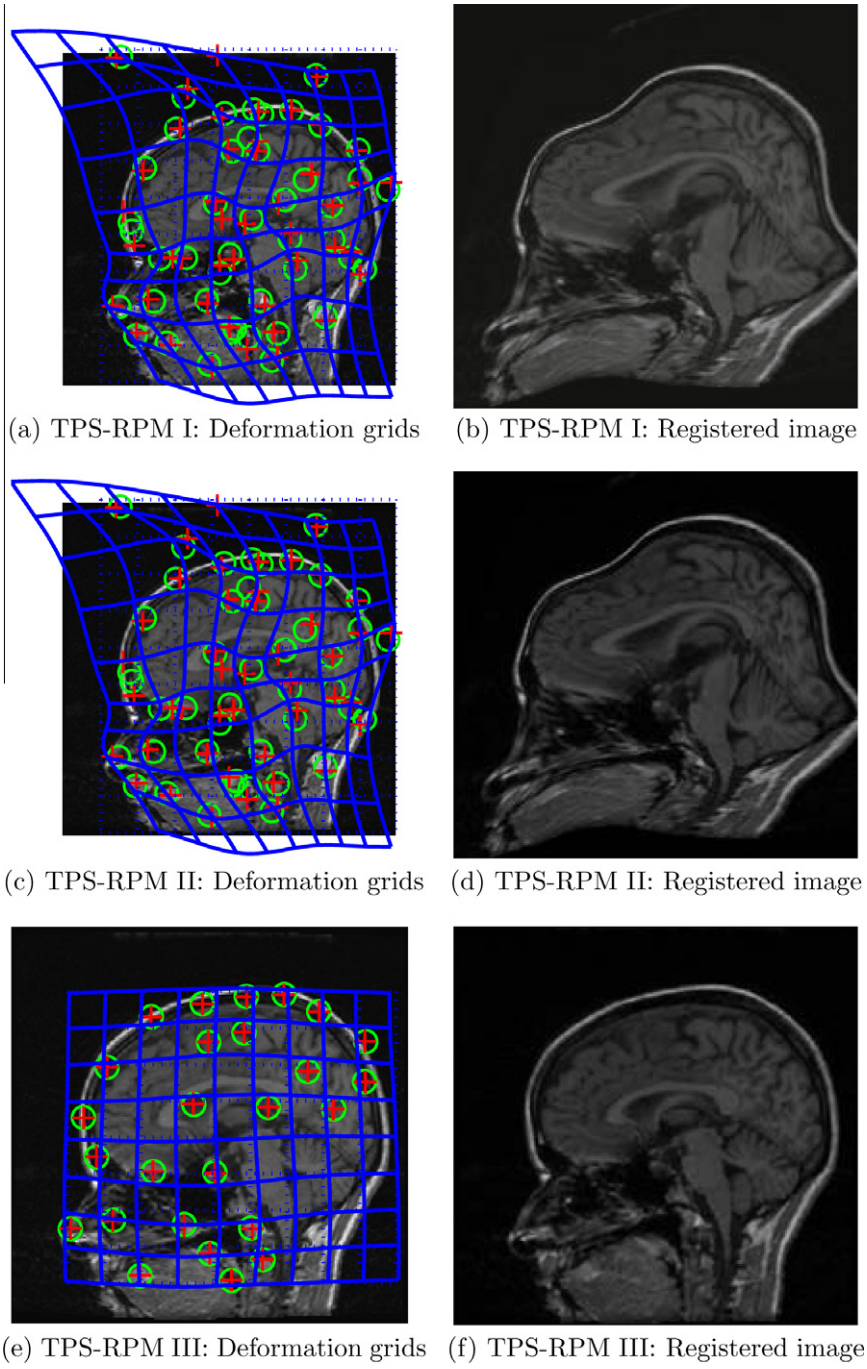

Fig. 5. Comparison of the registration results from the original TPS-RPM algorithm without intensity information (TPS-RPM I), revised TPS-RPM algorithm without intensity information (TPS-RPM II) and with intensity information (TPS-RPM III).

are referred to Lin et al. (2003) for details in using the curvature information.

The second example is an image registration, and the major purpose of this example is to demonstrate the importance of including the intensity information for registration when a great amount of outliers existing in both point sets simultaneously. We use the brain image (reference image) shown in Fig. 4(a) for example. The randomly TPS-deformed image (alignment image), the corresponding features, and the ground truth deformation grids are shown in Fig. 4(b). We will register the alignment image to the reference image using the simulated corresponding features and randomly generated outliers. We randomly add outliers to both the moving point set (extracted from the reference image) and the fixed point set (extracted from the alignment image) and then perform the registration using (1) the original TPS-RPM algorithm without including the intensity information (TPS-RPM I), (2) the revised TPS-RPM algorithm with double-sided outlier handler but not including the intensity information (TPS-RPM II), and (3) the revised TPS-RPM algorithm with double-sided outlier handler and including the intensity information (TPS-RPM III). The registration results are shown in Fig. 5. It shows that without including the intensity information, both TPS-RPM I and TPS-RPM II algorithms are not able to identify those outliers in two point sets that are close to each other in the Euclidean space after the transformation, even if the TPS-RPM II possesses the double-sided outlier handling property. To distinguish them, one has to include additional information, such as the

Table 1
Comparison of the GCC value between the reference image and the alignment images before and after the registration for a MR lung series.

Time point	T1	T2	T3
Before registration	0.9709	0.9559	0.9748
After registration	0.9848	0.9780	0.9847

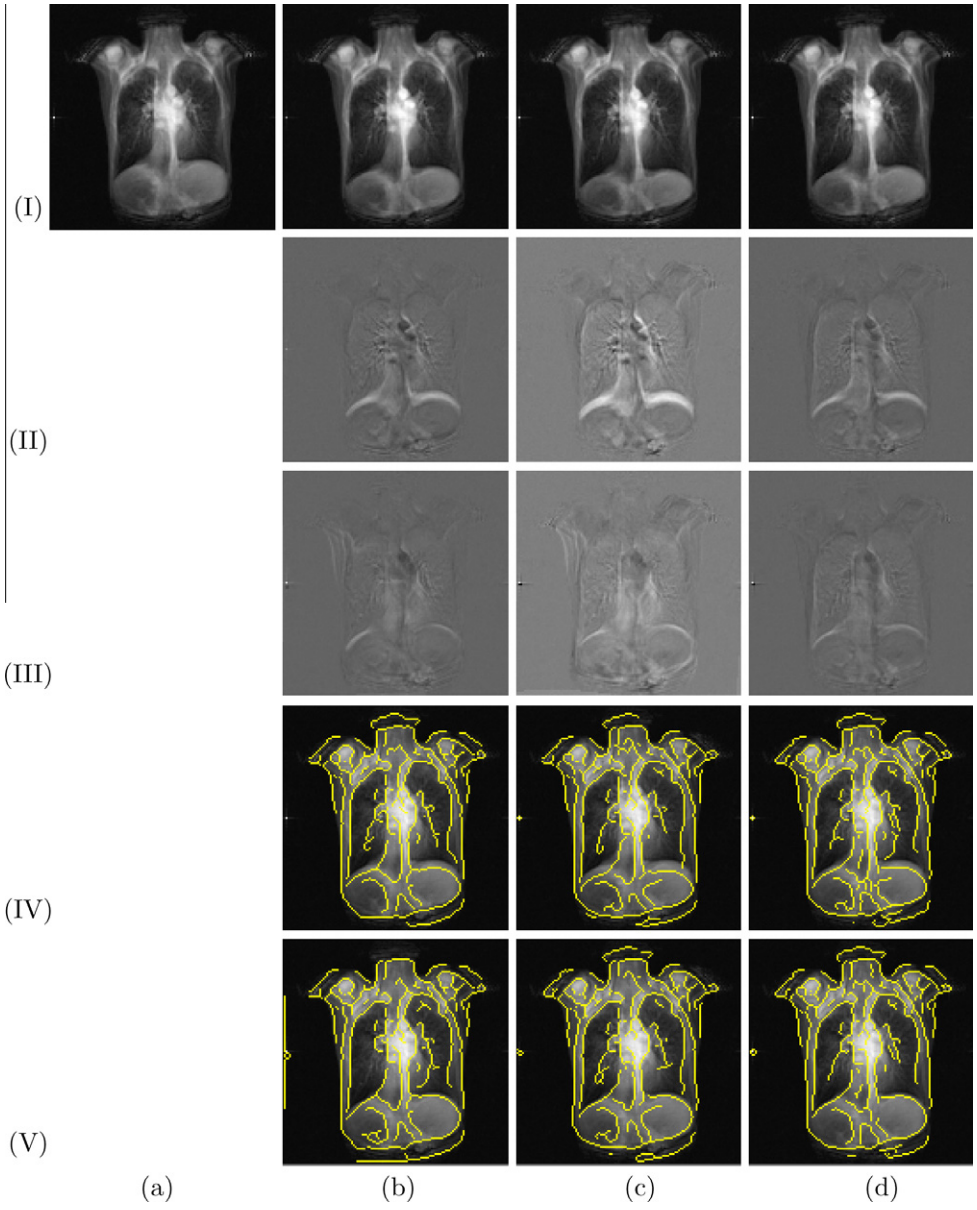

Fig. 6. Registration of a sequence of lung images at different time points, with (I.a) the reference image (exhale) and (I.b–d) the alignment images in T1, T2 and T3 time points (inhale), respectively. (II.b–d) The alignment images subtracted from the reference image. (III.b–d) The registered images subtracted from the reference image. (IV.b–d) Edges from the alignment images superimposed on the reference image. (V.b–d) Edges from the registered images superimposed on the reference image.

intensity information, to determine the correspondence. By including the intensity information for the point/feature matching, all those outliers have been detected and rejected, and the true deformation is recovered (see Fig. 5(e) and (f)).

In the third example we apply the image registration in Section 4 to registering a sequence of MR lung images. Four lung images (128×128) taken at different times in a breathing cycle are identified, three of which are registered to the fourth one at exhale breathing state. We compute the global cross-correlation coefficient (GCC) between the reference image and the alignment images before and after the registration, respectively. These values are reported in Table 1. We can see from the table that the GCC values are improved after the registration for all three alignment images. Fig. 6 shows the registration of the three alignment images at inhale breathing states. The subtraction map as well as the edge overlay of alignment image and the corresponding ones of registered image are shown in Fig. 6 for comparison. We can see that the difference between the registered image and the reference image is much smaller than that between the alignment image and the reference image, which implies a good registration.

6. Conclusion

To conclude, we reviewed the TPS-RPM algorithm and remarked its inherent limited performance when outliers are present in both point sets simultaneously. We demonstrated a double-sided outlier handling approach to overcome this limitation and proved the equivalence of this approach to the original formulation of the point matching problem within a rigorous mathematical framework, thus providing the theoretical foundation for the double-sided outlier handing property in the TPS-RPM algorithm. For a practical application, we also extended the TPS-RPM algorithm to non-rigid image registration by incorporating the intensity information for efficient outlier detection and rejection. These theoretical analysis and mathematical development have been validated by our experiments. Our work might shed light on the new development of more sophisticated non-rigid registration methods that integrate advanced outlier detection mechanism in the future.

References

Belongie, S., Malik, J., Puzicha, J., 2002. Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. 24 (4), 509–522.

Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. IEEE Trans. Pattern Anal. Mach. Intell. 14 (2), 239–256.

Boughorbel, F., Mercimek, M., Koschan, A., Abidi, M., 2010. A new method for the registration of three-dimensional point-sets: The Gaussian fields framework. Image Vision Comput. 28 (1), 124–137.

Chui, H., Rangarajan, A., 2000. A feature registration framework using mixture models. In: Proc. IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, pp. 190–197.

Chui, H., Rangarajan, A., 2003. A new point matching algorithm for non-rigid registration. Computer Vision and Image Understanding 89 (2-3), 114–141.

Cross, A.D.J., Hancock, E.R., 1998. Graph matching with a dual-step EM algorithm. IEEE Trans. Pattern Anal. Mach. Intell. 20 (11), 1236–1253.

Gerogiannis, D., Nikou, C., Likas, A., 2009. The mixtures of Student's *t*-distributions as a robust framework for rigid registration. Image Vision Comput. 27 (9), 1285–1294.

Huang, X., Sun, Y., Metaxas, D., Sauer, F., Xu, C., 2004. Hybrid image registration based on configural matching of scale-invariant salient region features. In: IEEE CVPR Workshop on Image and Video Registration, vol. 11, p. 167.

Kadir, T., Brady, M., 2001. Saliency, scale and image description. Internat. J. Comput. Vision 45 (2), 83–105.

Lin, N., Papademetris, X., Sinusas, A., Duncan, J.S., 2003. Analysis of left ventricular motion using a general robust point matching algorithm. In: Proc. Medical Image Computing and Computer-Assisted Intervention-Part I, pp. 556–563.

Maintz, J., Viergever, M., 1998. A survey of medical image registration. Med. Image Anal. 2 (1), 1–36.

Papadimitriou, C., Steiglitz, K., 1998. Combinatorial Optimization: Algorithms and Complexity. Dover Publications, Mineola, NY.

Rangarajan, A., Chui, H., Bookstein, F., 1997. The softassign procrustes matching algorithm. In: Proc. Information Processing in Medical Imaging, pp. 29–42.

Rohr, K., Fornefett, M., Stiehl, H.S., 1999. Approximating thin-plate splines for elastic registration: Integration of landmark errors and orientation attributes. In: Information Processing in Medical Imaging. Spring-Verlag, pp. 252–265.

Rohr, K., Stiehl, H.S., Sprengel, R., Buzug, T.M., Weese, J., Kuhn, M.H., 2001. Landmark-based elastic registration using approximating thin-plate splines. IEEE Trans. Med. Imag. 20 (6), 526–534.

Sinkhorn, R., 1964. A relationship between arbitrary positive matrices and doubly stochastic matrices. Ann. Math. Stat. 35 (2), 876–879.

Taron, M., Paragios, N., Jolly, M.-P., 2009. Registration with uncertainties and statistical modeling of shapes with variable metric kernels. IEEE Trans. Pattern Anal. Mach. Intell. 31 (1), 99–113.

Wahba, G., 1990. Spline Models for Observational Data. SIAM, Philadelphia, PA.

Wang, Y., 1998. Smoothing spline models with correlated random errors. J. Amer. Statist. Assoc. 93 (441), 341–348.

Wu, G., Qi, F., Shen, D., 2005. Learning best features for deformable registration of MR brains. In: Proc. Medical Image Computing and Computer-Assisted Intervention-Part II, pp. 179–187.

Yang, J., Williams, J.P., Sun, Y., Blum, R.S., Xu, C., 2006. Non-rigid image registration using geometric features and local salient region features. In: Proc. Computer Vision and Pattern Recognition, vol. 1, pp. 825–832.

Yuille, A.L., Kosowsky, J.J., 1994. Statistical physics algorithms that converge. Neural Comput. 6 (3), 341–356.

Zheng, Y., Doermann, D., 2006. Robust point matching for nonrigid shapes by preserving local neighborhood structures. IEEE Trans. Pattern Anal. Mach. Intell. 28 (4).

Zitová, B., Flusser, J., 2003. Image registration methods: A survey. Image Vision Comput. 21 (11), 977–1000.