ACADEMIC
PRESS

Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT®

Computer Vision and Image Understanding 89 (2003) 114–141

Computer Vision
and Image
Understanding

www.elsevier.com/locate/cviu

A new point matching algorithm for non-rigid registration

Haili Chui[a] and Anand Rangarajan[b,*]

[a] R2 Technologies, Sunnyvale, CA 94087, USA
[b] Department of Computer and Information Science and Engineering, University of Florida, Gainesville, FL 32611-6120, USA

Received 5 February 2002; accepted 15 October 2002

Abstract

Feature-based methods for non-rigid registration frequently encounter the correspondence problem. Regardless of whether points, lines, curves or surface parameterizations are used, feature-based non-rigid matching requires us to automatically solve for correspondences between two sets of features. In addition, there could be many features in either set that have no counterparts in the other. This outlier rejection problem further complicates an already difficult correspondence problem. We formulate feature-based non-rigid registration as a non-rigid point matching problem. After a careful review of the problem and an in-depth examination of two types of methods previously designed for rigid robust point matching (RPM), we propose a new general framework for non-rigid point matching. We consider it a general framework because it does not depend on any particular form of spatial mapping. We have also developed an algorithm—the TPS–RPM algorithm—with the thin-plate spline (TPS) as the parameterization of the non-rigid spatial mapping and the softassign for the correspondence. The performance of the TPS–RPM algorithm is demonstrated and validated in a series of carefully designed synthetic experiments. In each of these experiments, an empirical comparison with the popular iterated closest point (ICP) algorithm is also provided. Finally, we apply the algorithm to the problem of non-rigid registration of cortical anatomical structures which is required in brain mapping. While these results are somewhat preliminary, they clearly demonstrate the applicability of our approach to real world tasks involving feature-based non-rigid registration.

[*] Corresponding author.
E-mail addresses: hchui@r2tech.com (H. Chui), anand@cise.ufl.edu (A. Rangarajan).

1077-3142/03/$ - see front matter © 2003 Published by Elsevier Science (USA).
doi:10.1016/S1077-3142(03)00009-2

Keywords: Registration; Non-rigid mapping; Correspondence; Feature-based; Softassign; Thin-plate splines (TPS); Robust point matching (RPM); Linear assignment; Outlier rejection; Permutation matrix; Brain mapping

1. Introduction

Feature-based registration problems frequently arise in the domains of computer vision and medical imaging. With the salient structures in two images represented as compact geometrical entities (e.g., *points*, *curves*, and *surfaces*), the registration problem is to find the optimum or a good suboptimal spatial transformation/mapping between the two sets of features. The point feature, represented by feature location is the simplest form of feature. It often serves as the basis upon which other more sophisticated representations (such as curves, surfaces) can be built. In this sense, it can also be regarded as the most fundamental of all features. However, feature-based registration using point features alone can be quite difficult.

primitives

One common factor is the noise arising from the processes of image acquisition and feature extraction. The presence of noise means that the resulting feature points cannot be exactly matched. Another factor is the existence of outliers—many point features may exist in one point-set that have no corresponding points (homologies) in the other and hence need to be rejected during the matching process. Finally, the geometric transformations may need to incorporate high dimensional non-rigid mappings in order to account for deformations of the point-sets. Consequently, a general point feature registration algorithm needs to address all these issues. It should be able to solve for the correspondences between two point-sets, reject outliers and determine a good non-rigid transformation that can map one point-set onto the other.

outlier 是没有对应点

1. correspondence;
2. outlier/noise
3. transformation

The need for non-rigid registration occurs in many real world applications. Tasks like template matching for hand-written characters in OCR, generating smoothly interpolated intermediate frames between the key frames in cartoon animation, tracking human body motion in motion tracking, recovering dynamic motion of the heart in cardiac image analysis and registering human brain MRI images in brain mapping, all involve finding the optimal transformation between closely related but different objects or shapes. It is such a commonly occurring problem that many methods have been proposed to attack various aspects of the problem. However, because of the great complexity introduced by the high dimensionality of the non-rigid mappings, all existing methods usually simplify the problem to make it more tractable. For example, the mappings can be approximated by articulated rigid mappings instead of being fully non-rigid. Restricting the point-sets to lie along curves, the set of correspondence can be constrained using the curve ordering information. Simple heuristics such as using nearest-neighbor relationships to assign correspondence (as in the iterated closest point algorithm [5]) have also been widely used. Though most methods tolerate a certain amount of noise, they normally assume that there are no outliers. These simplifications may alleviate the difficulty of the matching problem, but they are not always valid. The non-rigid point matching problem, in this sense, still remains unsolved. Motivated by these observations, we feel that there is a need for a new point matching algorithm that can solve for non-rigid mappings as well as the correspondences in the presence of noise and outliers.

Our approach to non-rigid point matching closely follows our earlier work on joint estimation of pose and correspondence using the softassign and deterministic annealing [9,18,31]. This work was developed within an optimization framework and resulted in a robust point matching (RPM) algorithm which was restricted to using affine and piecewise-affine mappings. Here, we extend the framework to include spline-based deformations and develop a general purpose non-rigid point matching algorithm. Furthermore, we develop a specific algorithm—TPS–RPM—wherein we adopt the thin-plate spline (TPS) [7,43] as the non-rigid mapping. The thin-plate spline is chosen because it is the only spline that can be cleanly decomposed into affine and non-affine subspaces while minimizing a bending energy based on the second derivative of the spatial mapping. In this sense, the TPS can be considered to be a natural non-rigid extension of the affine map.

The rest of the paper is organized as follows. A detailed review of related work is presented in the following section. The optimization approach culminating in the TPS–RPM algorithm is developed in Section 3. This is followed by validation experiments using synthetic examples and a preliminary evaluation of the algorithm in the field of brain mapping. We then conclude by pointing out possible extensions of the present work.

2. Previous work

There are two unknown variables in the point matching problem—the correspondence and the transformation. While solving for either variable without information regarding the other is quite difficult, an interesting fact is that solving for one variable once the other is known is much simpler than solving the original, coupled problem.

2.1. Methods that solve only for the spatial transformation

The method of moments [21] is a classical technique which tries to solve for the transformation without ever introducing the notion of correspondence. The center of mass and the principal axis can be determined by moment analysis and used to align the point sets. A more sophisticated technique is the Hough Transform [3,38]. The transformation parameter space is divided into small bins, where each bin represents a certain configuration of transformation parameters. The points are allowed to vote for all the bins and the transformation bin which get the most votes is chosen. There are other methods, e.g., tree searches [2,20], the Hausdorff distance [25], geometric hashing [24,26] and the alignment method [42]. These methods work well for rigid transformations. However, these methods cannot be easily extended to the case of non-rigid transformations where the number of transformation parameters often scales with the cardinality of the data set. We suspect that this is the main reason why there is relatively a dearth of literature on non-rigid point matching despite a long and rich history on rigid, affine and projective point matching [19].

2.2. Methods that solve only for the correspondence

A different approach to the problem focuses on determining the right correspondence. As a brute force method, searching through all the possible point correspondences is obviously not possible. Even without taking into consideration the outliers, this leads to a combinatorial explosion. Certain measures have to be taken to prune the search in correspondence space. There are three major types of methods designed to do just that.

The first type of method, called dense feature-based methods, tries to group the feature points into higher level structures such as *lines, curves, or surfaces* through *object parameterization*. Then, the allowable ways in which the object can deform is specified [28,33]. In other words, curves and/or surfaces are first fitted to features extracted from the images [14,28,40,41]. Very often, a common parameterized coordinate frame is used in the fitting step, thereby alleviating the correspondence problem. The advantages and disadvantages can be clearly seen in the curve matching case. With the extra curve ordering information now available, the correspondence space can be drastically reduced making the correspondence problem much easier. On the other hand, the requirement of such extra information poses limitations for these methods. These methods work well only when the curves to be matched are reasonably smooth. Also, the curve fitting step that precedes matching is predicated on good feature extraction. Both curve fitting and feature extraction can be quite difficult when the data are noisy or when the shapes involved become complex. Most of these methods cannot handle *multiple* curves or *partially occluded* curves.

The second type of method works with more sparsely distributed point-sets. The basic idea is that while a point-set of a certain *shape* is non-rigidly deforming, different points at different locations can be assigned different attributes depending on their ways of movement. These movement attributes are used to distinguish the points and determine their correspondences. Following [12,34,35], the modal matching approach in [33] uses a mass and stiffness matrix that is built up from the Gaussian of the distances between point features in either point-set. The *mode shape vectors* are obtained as the eigenvectors of the *decoupled dynamic equilibrium equations*, which are defined using the above matrices. The correspondence is computed by comparing each point's relative participation in the eigen-modes. One major limitation of these algorithms is that they cannot tolerate outliers. Outliers can cause serious changes to the deformation modes invalidating the resulting correspondences. The accuracy of the correspondence obtained by just comparing the eigen-modes may also be limited.

The third type of approach recasts point matching as inexact, weighted graph matching [36]. Here, the *spatial relationships* between the points in each set are used to constrain the search for the correspondences. In [13], such interrelationships between the points is taken into account by building a graph representation from Delaunay triangulations. An expectation–maximization (EM) algorithm is used to solve the resulting graph matching optimization problem. However, the spatial mappings are restricted to be affine or projective. As we shall see, our approach in this work shares many similarities with [13]. In [1], decomposable graphs are hand-designed for deformable template matching and minimized with dynamic programming. In [27], a maximum clique approach is used to match relational sulcal graphs. In either case, there is no direct relationship between the deformable model and the graph used. The graph definition is also a common problem because interrelationships, attributes and link type information can be notoriously brittle and context dependent. In fact, the graphs in [1,27] are hand designed.

2.3. Methods that solve for both the correspondence and the transformation

Solving for the correspondence or the transformation alone seems rather difficult, if not impossible. Note that it is much easier to estimate the non-rigid transformation once the correspondence is given. On the other hand, knowledge of a reasonable spatial transformation can be of considerable help in the search for correspondence. This observation points to another way of solving the point matching problem—alternating estimation of correspondence and transformation.

The iterative closest point (ICP) algorithm [5] is the simplest of these methods. It utilizes the *nearest-neighbor* relationship to assign a binary correspondence at each step. This estimate of the correspondence is then used to refine the transformation, and vice versa. It is a very simple and fast algorithm which is guaranteed to converge to a local minimum. Under the assumption that there is an adequate set of initial poses, it can become a global matching tool for rigid transformations. Unfortunately, such an assumption is no longer valid in the case of non-rigid transformations, especially when then the deformation is large. ICP's crude way of assigning correspondence generates a lot of local minima and does not usually guarantee that the correspondences are one-to-one. Its performance degenerates quickly with outliers, even if some robustness control is added [9,31].

Recognizing these drawbacks of treating the correspondence as strictly a binary variable, other approaches relax this constraint and introduce the notion of "fuzzy" correspondence. There are mainly two kinds of approaches. In [13,22,44] a *probabilistic* approach is used. Point matching is modeled as a probability density estimation problem using a Gaussian mixture model. The well-known EM algorithm is used to solve the matching problem. When applied to the point matching problem, the E-step basically estimates the correspondence under the given transformation, while the M-step updates the transformation based on the current estimate of the correspondence. In [44], the outliers are handled through a uniform distribution in addition to the Gaussian mixture model. A similar digit recognition algorithm proposed in [22] models handwritten digits as deformable B-splines. Such a B-spline as well as an affine transformation are solved to match a digit pair. However, both [13,44] only solve for rigid transformations. Designed for character recognition, the approach in [22] assumes that the data points can be represented by B-splines and does not address the issue of matching arbitrary point patterns. One other common problem shared by all the probabilistic approaches is that they do not enforce one-to-one correspondence. In our previous work [9,18,31], point matching is modeled as a joint linear assignment-least squares *optimization* problem—a combination of two classical problems. Two novel techniques, namely deterministic annealing and softassign, were used to solve the joint optimization problem. Softassign used within deterministic annealing guarantees one-to-one correspondence. The resulting algorithm is actually quite similar to the EM algorithm.

Another approach recently proposed in Belongie et al. [4] adopts a different probabilistic strategy. A new shape descriptor, called the "shape context," is defined for correspondence recovery and shape-based object recognition. For each point chosen, lines are drawn to connect it to all other points. The length as well as the orientation of each line are calculated. The distribution of the length and the orientation for all lines (they are all connected to the first point) is estimated through histogramming. This distribution is used as the shape context for the first point. Basically, the shape context captures the distribution of the relative positions between the currently chosen point and all other points. The shape context is then used as the shape attribute for the chosen point. The correspondence can then be decided by comparing each point's attributes in one set with the attributes in the other. Since attributes and not relations are compared, the search for correspondence can be conducted much more easily. Or in more technical terms, the correspondence is obtained by solving a relatively easier bipartite matching problem rather than a more intractable (NP-complete) graph matching problem [16]. After the correspondences are obtained, the point set is warped and the method repeated. Designed with the task of object recognition in mind, this method has demonstrated promising performance in matching shape patterns such as hand-written characters. However, it is unclear how well this algorithm works in a registration context. Note that from a shape classification perspective, only a distance measure needs to be computed for the purposes of indexing. In contrast, in a registration setting, an accurate estimation of the spatial mapping is paramount. In addition, the convergence properties of this algorithm are unclear since there is no global objective function that is being minimized. Nonetheless, shape context may prove to be a useful way of obtaining attribute information which in turn helps in disambiguating correspondence.

2.4. The choice of the non-rigid transformation

As mentioned above, we choose the TPS to parameterize the non-rigid transformation. The TPS was developed in Wahba [43] as a general purpose spline tool which generates a smooth functional mapping for supervised learning. The mapping is a single closed-form function for the entire space. The smoothness measure is defined as the sum of the squares of the second derivatives of the mapping function over the space. Bookstein [7] pioneered the use of TPS to generate smooth spatial mappings between two sets of points with known one-to-one correspondences (landmarks) in medical images. Due to the limitation of known correspondence, use of the TPS has been restricted. The primary goal of this paper is to remove the limitation of known correspondence.

3. Point matching as joint estimation of correspondence and the spatial mapping

3.1. A binary linear assignment-least squares energy function

Suppose we have two point-sets V and X (in $\Re^2$or in $\Re^3$) consisting of points $\{v_a,\ a = 1, 2, \ldots, K\}$ and $\{x_i,\ i = 1, 2, \ldots, N\}$, respectively. For the sake of simplicity, we will assume for the moment that the points are in 2D.

We represent the non-rigid transformation by a general function f. A point v_a is mapped to a new location $u_a = f(v_a)$. The whole transformed point-set V is then U or $\{u_a\}$. We use a *smoothness* measure to place appropriate constraints on the mapping. To this end, we introduce an operator L and our chosen smoothness measure is $||Lf||^2$. We discuss this issue in greater detail in Section 4 where we specialize f to the thin-plate spline.

We now motivate the energy function used for non-rigid point matching. We would like to match the point-sets as closely as possible while rejecting a reasonable fraction of the points as outliers. The correspondence problem is cast as a linear assignment problem [29], which is augmented to take outliers into account. The benefit matrix in the linear assignment problem depends on the non-rigid transformation. More formally, our goal in this work is to minimize the following *binary linear assignment-least squares* energy function

$$\min_{Z,f} E(Z,f) = \min_{Z,f} \sum_{i=1}^{N} \sum_{a=1}^{K} z_{ai}||x_i - f(v_a)||^2 + \lambda ||Lf||^2 - \zeta \underbrace{\sum_{i=1}^{N} \sum_{a=1}^{K} z_{ai}}_{\text{控制rejection}} \qquad (1)$$

除了Lf这一项其他一样

subject to $\sum_{i=1}^{N+1} z_{ai} = 1$ for $a \in \{1, 2, \ldots K\}$, $\sum_{a=1}^{K+1} z_{ai} = 1$ for $i \in \{1, 2, \ldots, N\}$, and $z_{ai} \in \{0, 1\}$. The matrix Z or $\{z_{ai}\}$ is the binary *correspondence matrix* [18,31] consisting of two parts. The inner $N \times K$ part of Z defines the correspondence. If a point v_a corresponds to a point x_i, $z_{ai} = 1$, otherwise $z_{ai} = 0$. The row and column summation constraints guarantee that the correspondence is one-to-one. The extra $N + 1$th row and $K + 1$th column of Z are introduced to handle the outliers. Once a point is rejected as an outlier, the extra entries will start taking non-zero values to satisfy the constraints, when the related inner entries all become zero. (An example of the correspondence matrix is given in Fig. 1.) The second term is our constraint on

z_{ai}	x_1	x_2	x_3	x_4	outlier
v_1	1	0	0	0	0
v_2	0	1	0	0	0
v_3	0	0	0	0	1
outlier	0	0	1	1	

Fig. 1. An example of the binary correspondence matrix. Points v_1 and v_2 correspond to x_1 and x_2, respectively, and the rest of the points are outliers. Note that the existence of an extra outlier row and outlier column makes it possible for the row and column constraints to always be satisfied.

the transformation. The third term is the robustness control term preventing rejection of too many points as outliers. The parameters λ and ζ are the weight parameters that balance these terms.

?为什么可以有rejection?

Posed in this manner, the point matching objective function in (1) consists of two interlocking optimization problems: a linear assignment discrete problem on the correspondence and a least-squares continuous problem on the transformation. Both problems have unique solutions when considered separately. It is their combination that makes the non-rigid point matching problem difficult. Since the correspondence (transformation) has an optimal solution when the transformation (correspondence) is held fixed, it is natural to consider an alternating algorithm approach. However, even if such an approach is used, solving for *binary* one-to-one correspondences (and outliers) at each step is not meaningful when the transformation is far away from the optimal solution. Consequently, we have adopted an alternating algorithm approach where the correspondences are not allowed to approach binary values until the transformation begins to converge to a reasonable solution. This is done with the help of two techniques—softassign and deterministic annealing, which have been previously used to solve such joint optimization problems [9,18,31]. It is also similar to the dual-step EM algorithm in [13] wherein a similar alternating algorithm approach is used. Note that in [13], there is no attempt to solve for non-rigid deformations.

3.2. Softassign and deterministic annealing

The basic idea of the softassign [9,18,31] is to relax the binary correspondence variable Z to be a continuous valued matrix M in the interval $[0,1]$, while enforcing the row and column constraints. The continuous nature of the correspondence matrix M basically allows fuzzy, partial matches between the point-sets V and X. From an optimization point of view, this fuzziness makes the resulting energy function better behaved [47] because the correspondences are able to improve gradually and continuously during the optimization without jumping around in the space of binary permutation matrices (and outliers). The row and column constraints are enforced via iterative row and column normalization [37] of the matrix M.

With this notion of fuzzy correspondence established, another very useful technique, deterministic annealing [18,45], can be used to directly control this fuzziness by adding an entropy term in the form of $T\sum_{i=1}^{N+1}\sum_{a=1}^{K+1} m_{ai}\log m_{ai}$ to the original assignment energy function (1). The newly introduced parameter T is called the temperature parameter. The name comes from the fact that as you gradually reduce T, the energy function is minimized by a process similar to physical annealing [45]. At higher temperatures, the entropy term forces the correspondence to be more fuzzy and hence becomes a factor in "convexifying" the objective function. The minima obtained at each temperature are used as initial conditions for the next stage as the temperature is lowered. This process of deterministic annealing is a useful heuristic in a variety of optimization problems [15,17,23].

3.3. A fuzzy linear assignment-least squares energy function

After introducing these two techniques, the original binary assignment-least squares problem is converted to the problem of minimizing the following *fuzzy assignment-least squares* energy function [9,18,31]

$$\begin{aligned} E(M,f) = {} & \sum_{i=1}^{N}\sum_{a=1}^{K} m_{ai}||x_i - f(v_a)||^2 + \lambda||Lf||^2 \\ & + T\sum_{i=1}^{N}\sum_{a=1}^{K} m_{ai}\log m_{ai} - \zeta\sum_{i=1}^{N}\sum_{a=1}^{K} m_{ai}, \end{aligned} \tag{2}$$

where m_{ai} still satisfies $\sum_{i=1}^{N+1} m_{ai} = 1$ for $a \in \{1,\ldots,K\}$ and $\sum_{a=1}^{K+1} m_{ai} = 1$ for $i \in \{1,\ldots,N\}$ with $m_{ai} \in [0,1]$. When the temperature T reaches zero, the fuzzy correspondence M becomes binary. This is a consequence of the following extension to the well-known Birkhoff theorem [6]: the set of doubly substochastic matrices is the convex hull of the set of permutation matrices and outliers.

The energy function in (2) can be minimized by an alternating optimization algorithm that successively updates the correspondence parameter M and the transformation function f while gradually reducing the temperature T. Such an algorithm has proven to be quite successful in the case of rigid point matching [18,31]. However, there are two issues that motivated us to examine this approach more carefully in the case of non-rigid point matching. First, even though the parameter ζ that weighs the robustness control term in (2) can be interpreted as a prior estimate of the percentage of outliers in both point-sets, there is no clear way to estimate this parameter. We have adopted a strategy wherein the outlier variable acts as a cluster center with a very large variance (T_0) and all points that cannot be matched are placed in this cluster. The outlier cluster for each point-set is placed at the center of mass. We note that this solution is not optimal and that more work is needed to reliably set the robustness parameter. Second, setting the parameter λ for the prior smoothness term can be difficult. Large values of λ greatly limit the range of non-rigidity of the transformation. On the other hand, the transformation can turn out to be too flexible at small values of λ, making the algorithm unstable. We have opted instead to gradually reduce λ via an annealing schedule (by setting $\lambda = \lambda^{\text{init}}T$, where λ^{init} is a constant). The basic idea behind this heuristic is that more global and rigid transformations should be first favored (which is the case for large λ) followed later by more local non-rigid transformations (which is the case for small λ). Finally, please note that each step in the alternating algorithm lowers the energy in (2).

3.4. The robust point matching (RPM) algorithm

The resulting RPM is quite similar to the EM algorithm and very easy to implement. It essentially involves a dual update process embedded within an annealing scheme. We briefly describe the algorithm.

Step 1: Update the correspondence: For the points $a = 1, 2, \ldots, K$ and $i = 1, 2, \ldots, N$,

$$m_{ai} = \frac{1}{T} \exp\left(-\frac{(x_i - f(v_a))^T (x_i - f(v_a))}{2T} \right) \tag{3}$$

and for the outlier entries $a = K + 1$ and $i = 1, 2, \ldots, N$,

$$m_{K+1,i} = \frac{1}{T_0} \exp\left(-\frac{(x_i - v_{K+1})^T (x_i - v_{K+1})}{2T_0} \right) \tag{4}$$

and for the outlier entries $a = 1, 2, \ldots K$ and $i = N + 1$,

$$m_{a,N+1} = \frac{1}{T_0} \exp\left(-\frac{(x_{N+1} - f(v_a))^T (x_{N+1} - f(v_a))}{2T_0} \right), \tag{5}$$

where v_{K+1} and x_{N+1} are the outlier cluster centers as explained above.

Run the iterated row and column normalization algorithm to satisfy the constraints until convergence is reached,

$$m_{ai} = \frac{m_{ai}}{\sum_{b=1}^{K+1} m_{bi}}, \quad i = 1, 2, \ldots, N, \tag{6}$$

$$m_{ai} = \frac{m_{ai}}{\sum_{j=1}^{N+1} m_{aj}}, \quad a = 1, 2, \ldots, K. \tag{7}$$

The number of iterations in row and column normalization are fixed and independent of the number of points. This is because from one iteration to the next, the correspondence matrix does not change much and hence the previous value can be used as an initial condition.

Step 2: Update the transformation: After dropping the terms independent of f, we need to solve the following least-squares problem,

$$\min_f E(f) = \min_f \sum_{i=1}^{N} \sum_{a=1}^{K} m_{ai} ||x_i - f(v_a)||^2 + \lambda T ||Lf||^2. \tag{8}$$

Including the outliers in the least—squares formulation is very cumbersome. We implemented a slightly simpler form as shown below:

$$\min_f E(f) = \min_f \sum_{a=1}^{K} ||y_a - f(v_a)||^2 + \lambda T ||Lf||^2, \tag{9}$$

where

$$y_a = \sum_{i=1}^{N} m_{ai} x_i. \tag{10}$$

The variable y_a can be regarded as our newly estimated positions of the point-set (within the set $\{x_i\}$) that corresponds to $\{v_a\}$. Extra bookkeeping is needed to check for outliers (if $\sum_{i=1}^{N} m_{ai}$ is too small) and eliminate them. The solution for this least-squares problem depends on the specific form of the non-rigid transformation. We will discuss the solution for one form—the thin-plate spline in the next section.

Annealing: An annealing scheme controls the dual update process. Starting at $T_{\text{init}} = T_0$, the temperature parameter T is gradually reduced according to a linear annealing schedule, $T^{\text{new}} = T^{\text{old}} \cdot r$ (r is called the annealing rate). The dual updates are repeated till convergence at each temperature. Then T is lowered and the process is repeated until some final temperature T_{final} is reached.

The parameter T_0 is set to the largest square distance of all point pairs. We usually set r to be 0.93 (normally between [0.9,0.99]) so that the annealing process is slow enough for the algorithm to be robust, and yet not too slow. For the outlier cluster, the temperature is always kept at T_0. Since the data is often quite noisy, matching them exactly to get binary one-to-one correspondences is not always desirable. So the final temperature T_{final} is chosen to be equal to the average of the squared distance between the nearest neighbors within the set of points which are being deformed. The interpretation is that at T_{final}, the Gaussian clusters for all the points will then barely overlap with one another.

3.5. Relationship to ICP

The iterated closest point (ICP) algorithm [5] is actually quite similar to our algorithm. As we discussed in Section 2, the main difference is that ICP depends on the nearest-neighbor heuristic and always returns a binary correspondence, which is not guaranteed to be one-to-one. However, it is close to the limiting case of our algorithm when annealing is replaced with quenching (starting T close to zero). Since T can be regarded as a search range parameter, it is not hard to see that the over-simplified treatment of correspondence in ICP makes it much more vulnerable to local minima. A second difference is that ICP relies on other heuristics to handle outliers. For example, outliers are rejected by setting a dynamic thresholding mechanism [14]. Distances between nearest point pairs are first fit to a Gaussian distribution. Points with distance measure values larger than the mean plus L (usually set to 3) times the standard deviation are then rejected as outliers. We implemented this improved version of ICP as a benchmark comparison for our algorithm.

4. The thin-plate spline and the TPS–RPM algorithm

To complete the specification of the non-rigid point matching algorithm, we now discuss a specific form of non-rigid transformation—the thin-plate spline [7,43]. The generation of a smoothly interpolated spatial mapping with adherence to two sets of landmark points is a general problem in spline theory. Once non-rigidity is allowed, there are an infinite number of ways to map one point-set onto another. The smoothness constraint is necessary because it discourages mappings which are too arbitrary. In other words, we can control the behavior of the mapping by choosing a specific smoothness measure, which basically reflects our prior knowledge. One of the simplest measures is the space integral of the square of the second order derivatives of the mapping function. This leads us to the thin-plate spline.

The TPS fits a mapping function $f(v_a)$ between corresponding point-sets $\{y_a\}$ and $\{v_a\}$ by minimizing the following energy function

$$E_{\mathrm{TPS}}(f) = \sum_{a=1}^{K} ||y_a - f(v_a)||^2 + \lambda \int\int \left[\left(\frac{\partial^2 f}{\partial x^2}\right)^2 + 2\left(\frac{\partial^2 f}{\partial x \partial y}\right)^2 + \left(\frac{\partial^2 f}{\partial y^2}\right)^2\right] \mathrm{d}x\, \mathrm{d}y. \tag{11}$$

Suppose the points are in 2D ($D = 2$). We use *homogeneous* coordinates for the point-set where one point y_a is represented as a vector $(1, y_{ax}, y_{ay})$. With a fixed regularization parameter λ, there exists a unique minimizer f which comprises two matrices d and w,

$$f(v_a, d, w) = v_a \cdot d + \phi(v_a) \cdot w, \tag{12}$$

where d is a $(D+1) \times (D+1)$ matrix representing the affine transformation and w is a $K \times (D+1)$ warping coefficient matrix representing the non-affine deformation. The vector $\phi(v_a)$ is related to the TPS *kernel*. It is a $1 \times K$ vector for each point v_a, where each entry $\phi_b(v_a) = ||v_b - v_a||^2 \log ||v_b - v_a||$. Loosely speaking, the TPS kernel contains the information about the point-set's internal structural relationships. When combined with the warping coefficients w, it generates a non-rigid warping. If we substitute the solution for f (12) into (11), the TPS energy function becomes,

$$E_{\mathrm{TPS}}(d, w) = ||Y - Vd - \Phi w||^2 + \lambda \operatorname{trace}(w^T \Phi w), \tag{13}$$

where Y and V are just concatenated versions of the point coordinates y_a and v_a, and Φ is a $(K \times K)$ matrix formed from the $\phi(v_a)$. Each row of each newly formed matrix comes from one of the original vectors. The decomposition of the transformation into a global affine and a local non-affine component is a very nice property of the TPS. Consequently, the TPS smoothness term in (13) is solely dependent on the non-affine components. This is a desirable property, especially when compared to other splines, since the global pose parameters included in the affine transformation are not penalized.

As it stands, finding least-squares solutions for the pair d, w by directly minimizing (13) is awkward. Instead, a QR decomposition [43] is used to separate the affine and non-affine warping space.

$$V = [Q_1 Q_2]\begin{pmatrix} R & \\ & 0 \end{pmatrix}, \tag{14}$$

where Q_1 and Q_2 are $K \times (D+1)$ and $N \times (K - D - 1)$ orthonormal matrices, respectively. The matrix R is upper triangular. With the QR decomposition in place, (13) becomes,

$$E_{\mathrm{TPS}}(\gamma, d) = ||Q_2^T Y - Q_2^T \Phi Q_2 \gamma||^2 + ||Q_1^T Y - Rd - Q_1^T \Phi Q_2 \gamma||^2 + \lambda \gamma^T Q_2^T \Phi Q_2 \gamma, \tag{15}$$

where $w = Q_2\gamma$ and γ is a $(K - D - 1) \times (D + 1)$ matrix. Setting $w = Q_2\gamma$ (which in turn implies that $V^T w = 0$) enables us to cleanly separate the warping into affine and non-affine subspaces. This is in sharp contrast with the solution in [7] where the subspaces are not separated.

The least-squares energy function in (15) can be first minimized w.r.t γ and then w.r.t. d. The final solution for w and d are,

$$\widehat{w} = Q_2\widehat{\gamma} = Q_2(Q_2^T \Phi Q_2 + \lambda I_{(K-D-1)})^{-1} Q_2^T Y \tag{16}$$

and

$$\widehat{d} = R^{-1}(Q_1^T V - \Phi\widehat{w}). \tag{17}$$

We call the minimum value of the TPS energy function obtained at the optimum $(\widehat{w}, \widehat{d})$ the "bending energy."

$$E_{\text{bending}} = \lambda \,\text{trace}[Q_2(Q_2^T \Phi Q_2 + \lambda I_{(K-D-1)})^{-1} Q_2^T Y Y^T]. \tag{18}$$

We have encountered a minor problem with the TPS. Because the affine transformation component is not penalized at all in the TPS energy function, unphysical reflection mappings which can flip the whole plane can sometimes happen. To somewhat alleviate this problem, we added a constraint on the affine mapping d by penalizing the residual part of d which is different from an identity matrix I. This is not an ideal solution, merely convenient. A decomposition of the affine into separate physical rotation, translation, scale and shear components would be a more principled solution. The actual TPS energy function we used is the following

$$E_{\text{TPS}}(d, w) = ||Y - Vd - \Phi w||^2 + \lambda_1 \,\text{trace}(w^T \Phi w) + \lambda_2 \,\text{trace}[d - I]^T [d - I]. \tag{19}$$

The solution for d and w is similar to the one for the standard TPS. For the reasons explained earlier, the weight parameters λ_1 and λ_2 both have annealing schedules ($\lambda_i = \lambda_i^{\text{init}} T$, $i = 1, 2$). To provide more freedom for the affine transformation, λ_2^{init} is set to be much smaller than λ_1^{init}. While this approach works in practice, it is inelegant. We are currently looking at the work in [32] for ways of introducing regularization parameters on both the affine and the deformation in a principled way.

Incorporating TPS into the general point matching framework leads to a specific form of robust non-rigid point matching—TPS–RPM. The dual updates and the temperature parameter setup have been discussed above. We explain the setup for the rest of the parameters. The parameter λ_1^{init} is set to 1 and λ_2^{init} is set to $0.01 \cdot \lambda_1^{\text{init}}$. The alternating update of the correspondence M and the transformation d, w is repeated five times after which they are usually close to convergence. We then decrease the temperature T and repeat the entire process. The implementations of the thin plate spline in 2D and in 3D are almost identical except that in 3D, a different kernel of the form $\phi_b(v_a) = ||v_b - v_a||$ is used [43]. The pseudo-code for the TPS–RPM algorithm follows.

The TPS–RPM Algorithm Pseudo-code:

```
Initialize parameters $T$, $\lambda_1$, and $\lambda_2$.
Initialize parameters $M$, $d$, and $w$.
```

```
Begin A: Deterministic Annealing.
    Begin B: Alternating Update.
    Step I: Update correspondence matrix $M$ using (3)–(5).
    Step II: Update transformation parameters $(d, w)$ using (9).
    End B
Decrease $T$, $\lambda_1$, and $\lambda_2$.
End A
```

5. Experiments

5.1. A simple 2D example

We noticed that the algorithm demonstrates a very interesting scale-space like behavior. Such behavior can be seen from a simple 2D point matching example demonstrated here in Figs. 2 and 3.

In the beginning, at a very high temperature T, the correspondence M is almost uniform. Then the estimated corresponding point-set $\{y_a = \sum_{i=1}^{N} m_{ai}x_i\}$is essentially very close to the *center of mass* of X. This helps us recover most of the translation needed to align the two point-sets. This can be seen from the first column in Fig. 3.

With slightly lowered temperatures as shown in the 2nd column of Fig. 3, the algorithm starts behaving very much like a principal axis method. Points in V are rotated and aligned with the major axis along which the points in X are most densely distributed.

Lowering the temperature even more, we observe that the algorithm finds more localized correspondences which makes it possible to capture the more detailed structures within the target point-set. Progressively more refined matching is shown in the 3rd, 4th, and 5th columns. The last column shows that the algorithm almost converges to a nearly binary correspondence at a very low temperature and the correct non-rigid transformation is fully recovered.

The algorithm is clearly attempting to solve the matching problem using a coarse to fine approach. Global structures such as the center of mass and principal axis are

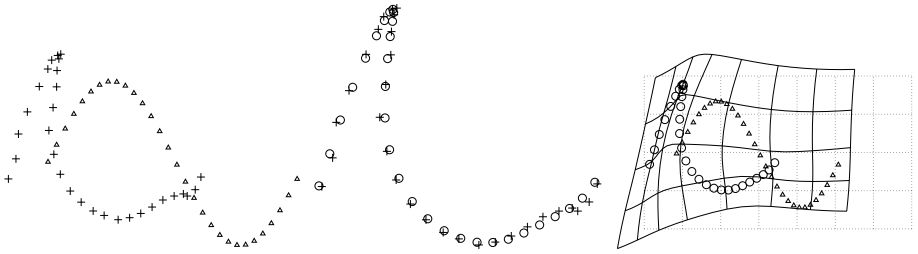


Fig. 2. A simple 2D example. *Left*: initial position. Two point sets V (triangles) and X (crosses). *Middle*: final position. U (transformed V) and X. *Right*: deformation of the space is shown by comparing the original regular grid (dotted lines) and its transformed version (solid lines).

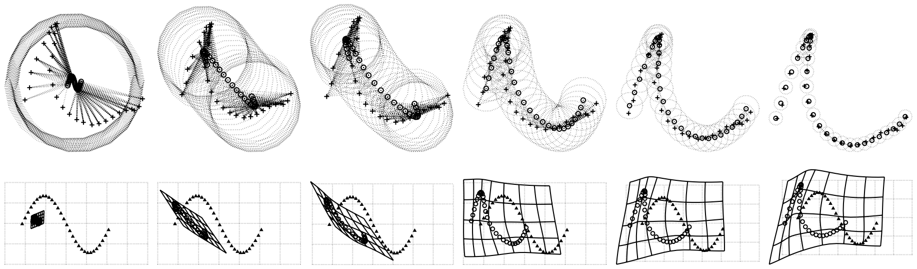


Fig. 3. Matching process. Each column shows the state of the algorithm at a certain temperature. *Top*: current correspondence between U (transformed V, circles) and X (crosses). The most significant correspondences ($m_{ai} > 1/K$) are shown as dotted links. A dotted circle is of radius $\sqrt{T}$ is drawn around each point in U to show the annealing process. *Bottom*: deformation of the space. Again dotted regular grid with the solid deformed grid. Original V (triangles) and U (transformed V, circles).

first matched at high temperature, followed by the non-rigid matching of local structures at lower temperatures. It is very interesting that during the process of annealing, this whole process occurs seamlessly and implicitly.

ICP's treatment of the correspondence as a binary variable determined by the nearest neighbor heuristic leads to significantly poorer performance when compared with RPM. The difference is most evident when outliers are involved. We have come across numerous examples where a small percentage of outliers would cause ICP to fail. To demonstrate the idea, here we tested both ICP and RPM on the same example as above with a single, extra outlier. Since we used an annealing schedule for the regularization parameters λ_1 and λ_2 in RPM, we also used it for ICP. The results are shown in Fig. 4. Even though there is only one outlier, ICP gets trapped and could not recover. The answer is obviously suboptimal. We see that RPM also gets affected by the outlier in an intermediate stage of the matching process. It eventually breaks free and converges to a better answer.

5.2. Evaluation of RPM and ICP through synthetic examples

To test RPM's performance, we ran a lot of experiments on synthetic data with different degrees of warping, different amounts of noise and different amounts of outliers and compared it with ICP.

After a *template* point-set is chosen, we apply a randomly generated non-rigid transformation to warp it. Then we add noise or outliers to the warped point-set to get a new *target* point-set. Instead of TPS, we use a different non-rigid mapping, namely Gaussian radial basis functions (RBF) [46] for the random transformation. The coefficients of the RBF were sampled from a Gaussian distribution with a zero mean and a standard deviation s_1. Increasing the value of s_1 generates more widely distributed RBF coefficients and hence leads to generally larger deformation. A certain percentage of outliers and random noise are added to the warped template to generate the target point-set. We then used both ICP and RPM to find the best TPS to map the template set onto the target set. The errors are computed as the

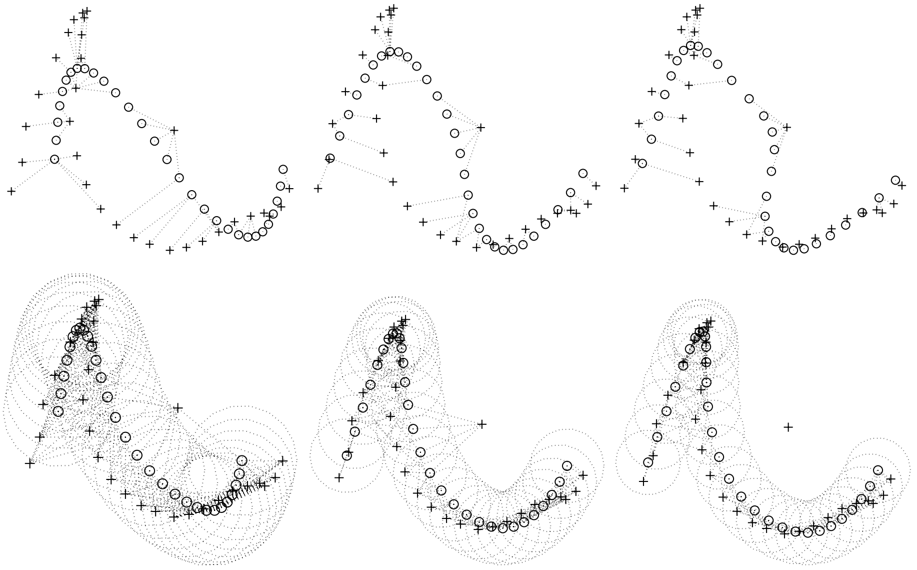


Fig. 4. An example with one outlier. *Top row*: sequential intermediate states of ICP. *Bottom row*: sequential intermediate states of RPM.

mean squared distance between the warped template using the TPS found by the algorithms and the warped template using the ground truth Gaussian RBF.

We conducted three series of experiments. In the first series of experiments, the template was warped through progressively larger degrees of non-rigid warpings. The warped templates were used as the target data without adding noise or outliers. The purpose is to test the algorithms' performance on solving different degrees of deformations. In the second series, different amounts of Gaussian noise (standard deviation s_2 from 0 to 0.05) were added to the warped template to get the target data. A medium degree of warping was used to warp the template. The purpose is to test the algorithms' tolerance of noise. In the third series, different amounts of random outliers (outlier to original data ratio s_3 ranging from 0 to 2) were added to the warped template. Again, a medium degree of warping was used. One hundred random experiments were repeated for each setting within each series.

We used two different templates. The first one comes from the outer contour of a tropical fish. The second one comes from a Chinese character (blessing), which is a more complex pattern. All three series of experiments were run on both templates.

Some examples of these experiments are shown in Figs. 5–10. The final statistics (error means and standard deviations for each setting) are shown in Fig. 11. ICP's performance deteriorates much faster when the examples becomes harder due to any of these three factors—degree of deformation, amount of noise or amount of outliers. The difference is most evident in the case of handling outliers. ICP starts failing in most of the experiments and gives huge errors once outliers are added. On the other hand, RPM seems to be relatively unaffected. It is very interesting that

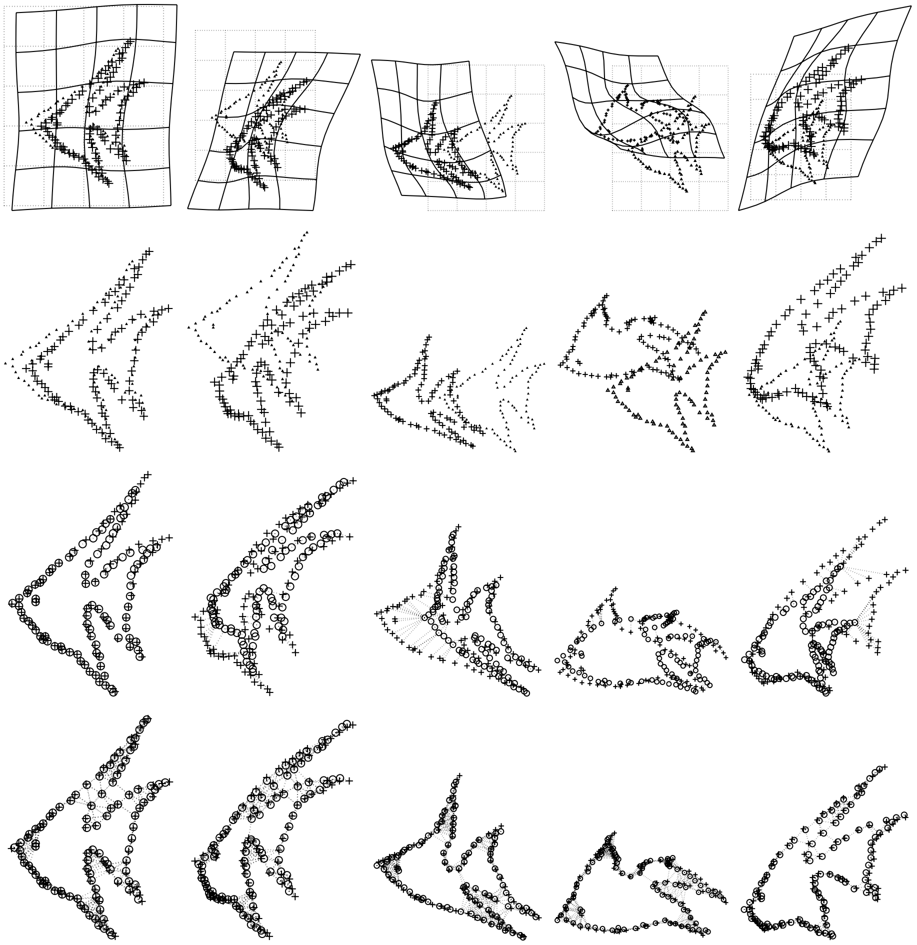


Fig. 5. Experiments on deformation. Each column represent one example. From left to right, increasing degree of deformation. *Top row*: warped template. *Second row*: template and target (same as the warped template). *Third row*: ICP results. *Bottom row*: RPM results.

this remains the case even when the numbers of outliers far exceeds the number of true data points.

5.3. Large deformation examples

From the synthetic experiments, it appears that RPM can recover large deformations to some extent. A natural question then is how large a deformation can RPM successfully recover. To answer this question, we conducted a new experiment.

We hand-drew a sequence of 12 caterpillar images. The caterpillar is first supine. Then it gradually bends its tail toward its head. Further into the sequence, the

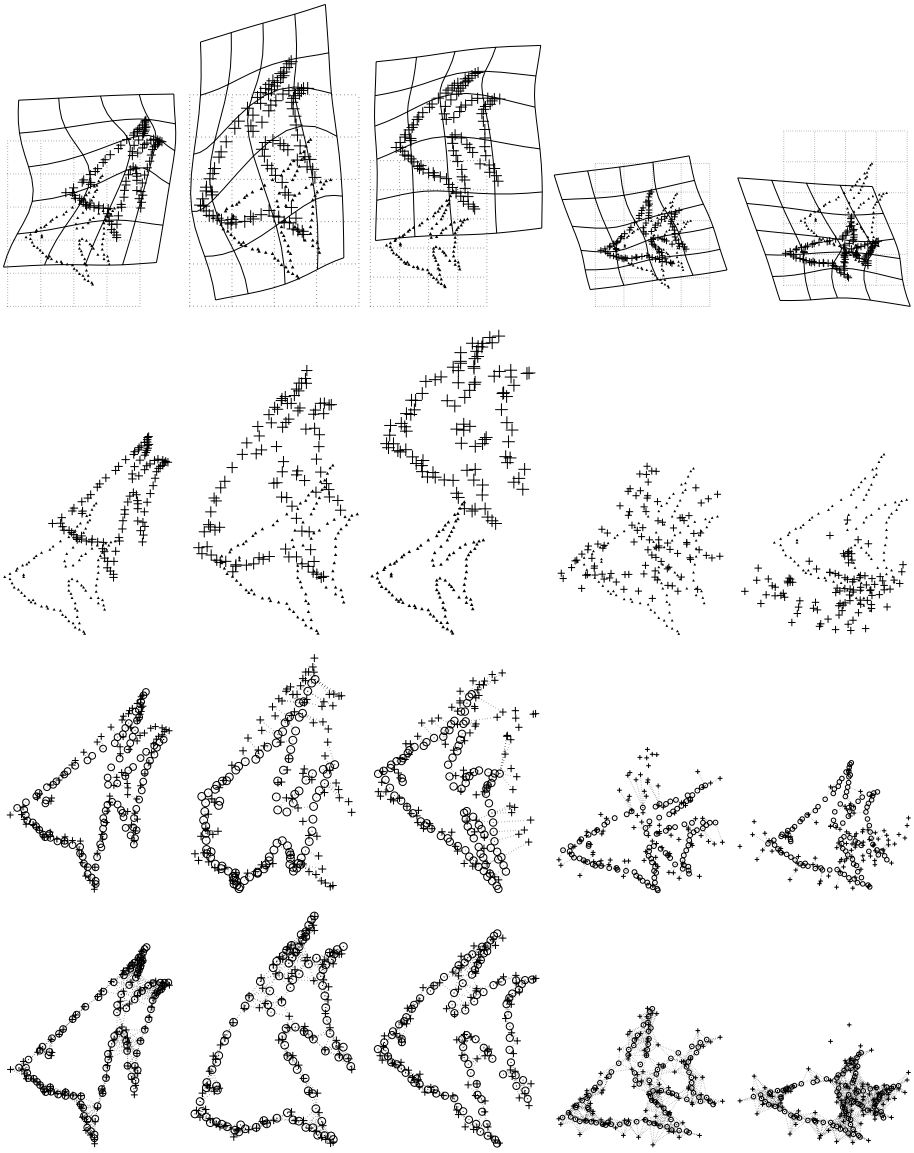


Fig. 6. Experiments on noise. From left to right, increasing amount of noise. *Top row*: warped template. *Second row*: template and target (warped template but with noise now). *Third row*: ICP results. *Bottom row*: RPM results.

bending increases. Points are extracted from each image in the sequence through thresholding. We then tried to match the first caterpillar point-set with the later deformed ones. The purpose is to test RPM's ability to track various degrees of deformation in the absence of outliers. The results of matching the first frame to the 5th,

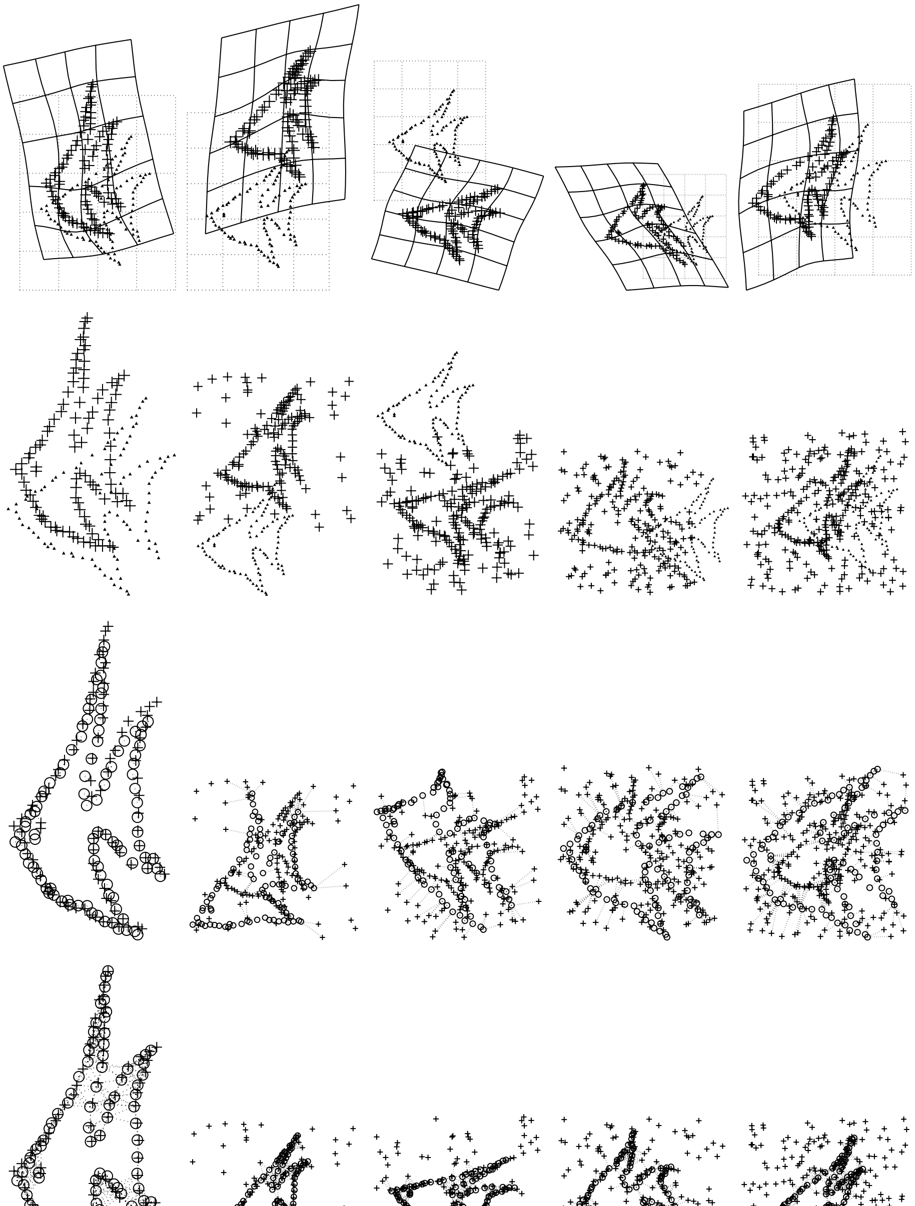


Fig. 7. Experiments on outliers. From left to right, increasing amount of outliers. *Top row*: warped template. *Second row*: template and target (warped template but with outliers now). *Third row*: ICP results. *Bottom row*: RPM results.

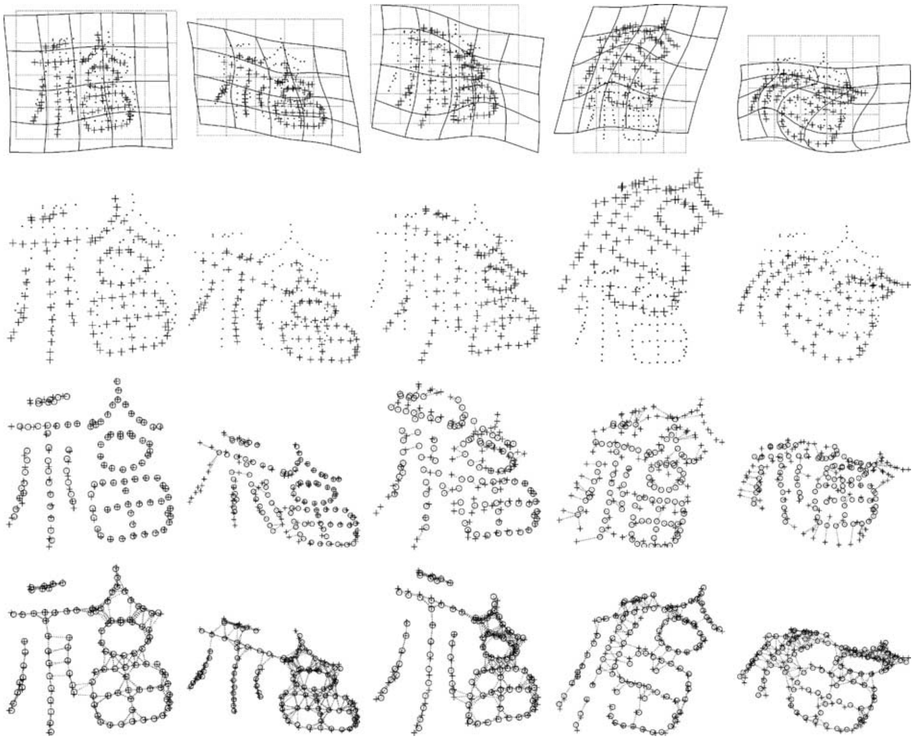


Fig. 8. Experiments on deformation. From left to right, increasing degree of deformation. *Top row*: warped template. *Second row*: template and target (same as the warped template). *Third row*: ICP results. *Bottom row*: RPM results.

the 7th, the 11th, and the 12th frame are shown in Fig. 12. We observed that up to frame eight, RPM is able to recover most of the desired transformation. Beyond that, it still tries to warp the points to get the best fit and ends up generating "strange" unphysical warpings. This shows the limitation of the thin-plate spline. We speculate that a diffeomorphic mapping would be able to provide a better solution when combined with the correspondence engine in RPM.

5.4. Brain mapping application

In brain mapping, one of the most important modules is a non-rigid anatomical brain MRI registration module. We now present preliminary experiments comparing TPS–RPM with other techniques. We wish to point out at this stage that we are merely trying to demonstrate the applicability of TPS–RPM to the non-rigid registration of cortical anatomical structures. Cortical sulci were traced using an SGI graphics platform [30] with a ray-casting technique that allows drawing in 3D space by projecting 2D coordinates of the tracing onto the exposed cortical surface. This is displayed on the left in Fig. 13. The interhemispheric fissure and 10 other major sulci

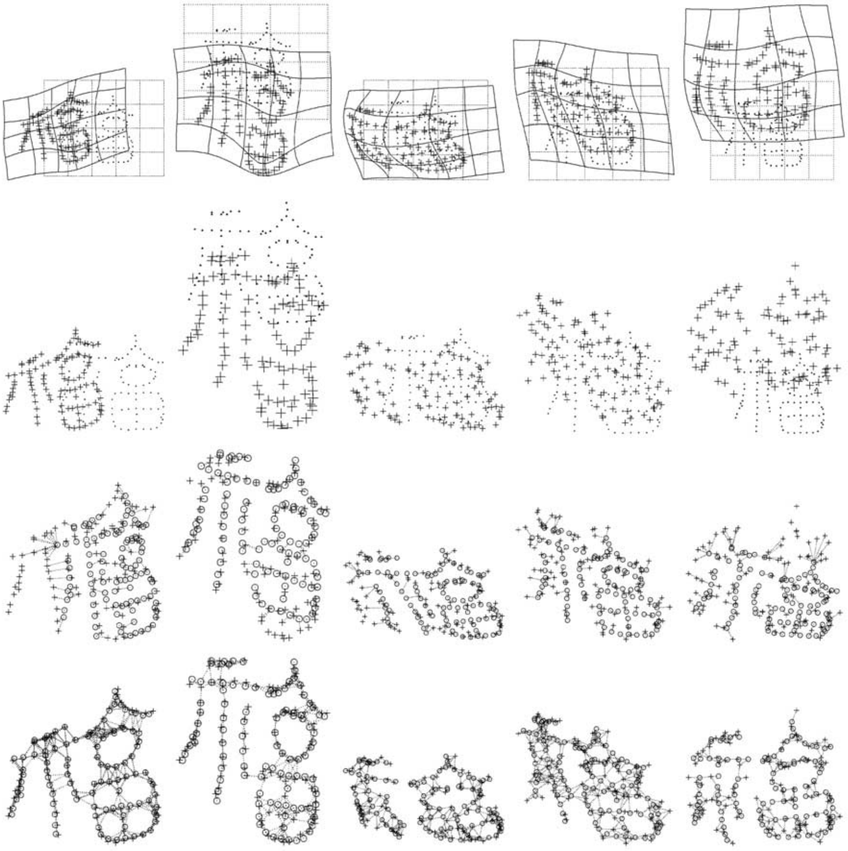


Fig. 9. Experiments on noise. From left to right, increasing amount of noise. *Top row*: warped template. *Second row*: template and target (warped template but with noise now). *Third row*: ICP results. *Bottom row*: RPM results.

(superior frontal, central, post-central, Sylvian, and superior temporal on both hemispheres) were extracted as point features. A sulcal point-set extracted from one subject is shown on the right in Fig. 13. Since there is no guarantee of consistency of minor cortical sulci across subjects, we restricted ourselves to major sulci. Also, representing sulci as curves is essentially wrong since they are 3D ribbon-like structures. And the TPS in 3D is poorly constrained by using 3D space curves as features. Despite these objections, we feel that this example demonstrates an application of TPS–RPM to a real-world problem. We are currently working on a much more realistic brain mapping application using 3D ribbon representations for sulci [11].

The original sulcal point-sets normally contain around 3000 points each. The point-set is first subsampled to have around 300 points by taking every 10th point. The original MRI volume's size is $106(X) \times 75(Y) \times 85(Z, \text{slices})$. With that in mind, it is reasonable to assume that the average distances between points before

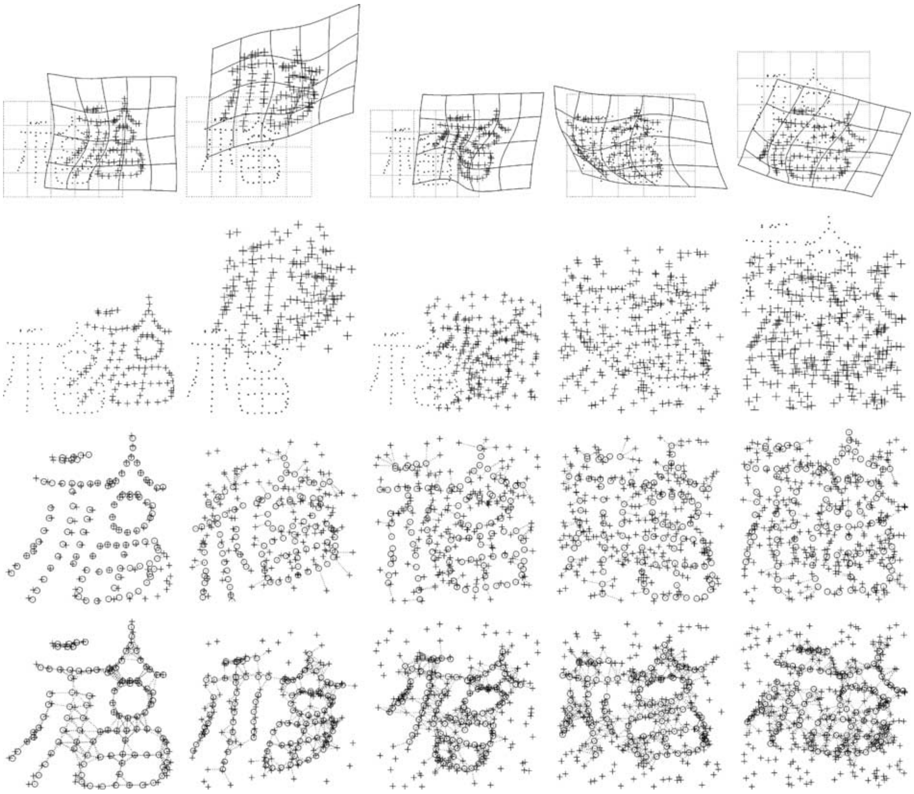


Fig. 10. Experiments on outliers. From left to right, increasing amount of outliers. *Top row*: warped template. *Second row*: template and target (warped template but with outliers now). *Third row*: ICP results. *Bottom row*: RPM results.

registration should be in the range of 10–100. We set our starting temperature to be roughly in the same range. After registration, we would expect the average distance between corresponding points to be within a few voxels (say, 1–10). Our final temperature should be slightly smaller. From these considerations, we set the RPM annealing parameters to be the following: $T_{\text{init}} = 50$, $T_{\text{final}} = 1$, $T_{\text{anneal-rate}} = 0.95$. The regularization parameters are linearly varied with the temperature as described above.

5.4.1. Sulcal point matching: a comparison between voxel-based matching (UMDS) and RPM

We applied TPS–RPM to five sulcal point-sets and compared it with three other methods which use affine (and piecewise-affine) transformations for brain registration.

We suspected that the voxel-based methods' performance would not be as satisfying as feature-based methods for sulcal alignment. To test this, we compared RPM with a voxel-based affine mapping method (UMDS) [39] which maximizes the

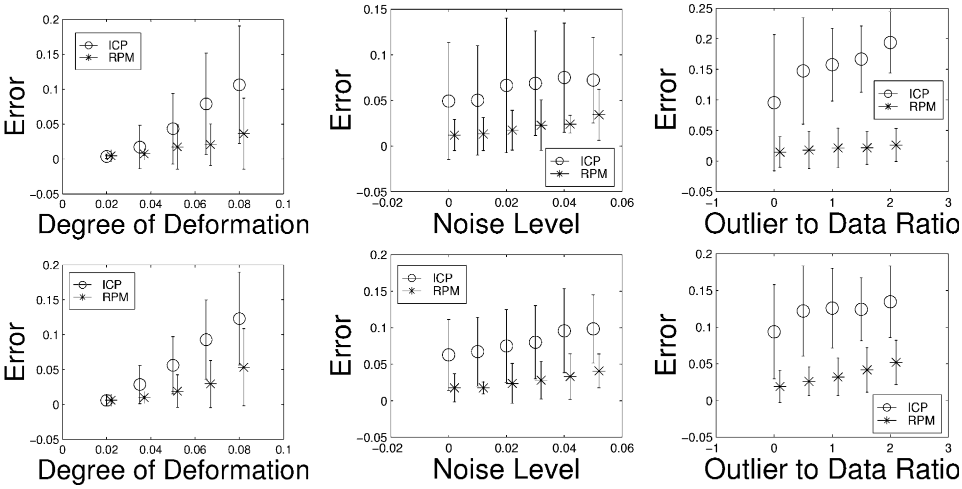


Fig. 11. Statistics of the synthetic experiments. *Top*: results using the fish template. *Bottom*: results using the Chinese character template. Note especially how RPM's errors increase much slower than ICP's errors in all cases.

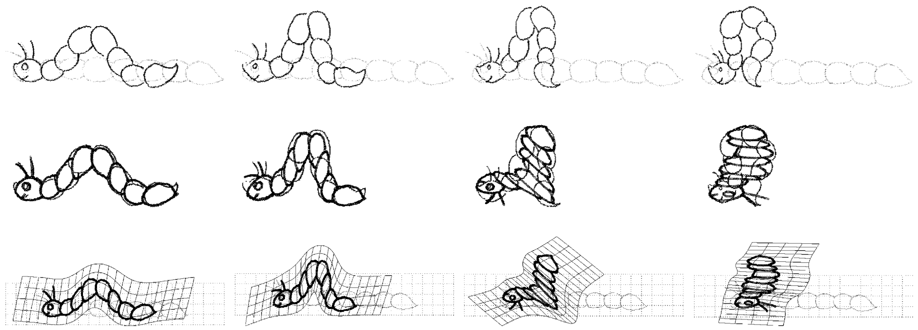
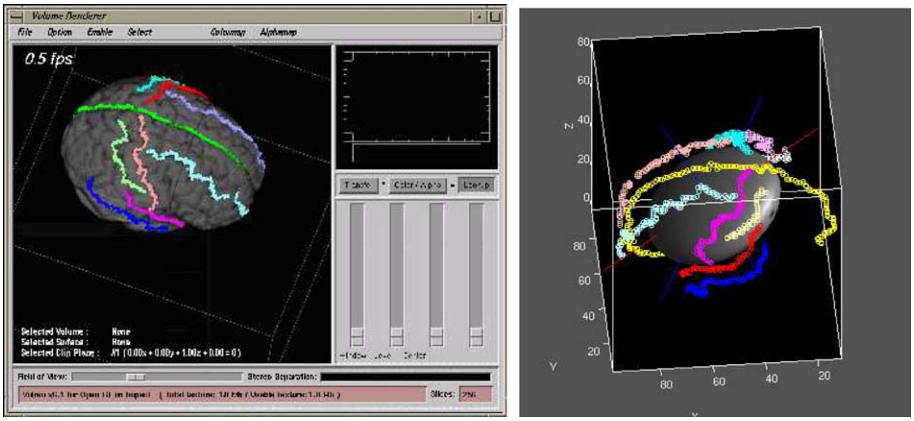


Fig. 12. Large deformation—Caterpillar example. From left to right, matching frame 1–5, 7, 11, and 12. *Top*: original location. *Middle*: matched result. *Bottom*: deformation found.

mutual information between the two volumes. The volumes were then matched using UMDS as described in [39] and the resulting spatial mapping was applied to the sulcal points. RPM was separately run on the sulcal point-sets, and the results from global affine, piecewise affine and thin-plate spline mappings are shown in Figs. 14 and 15. Since we register every brain to the first one, after registration, the minimum distance from each sulcal point in the current brain to the first is calculated.

The above comparison of RPM with the voxel-based UMDS approach shows that RPM can improve upon a voxel-based approach at the sulci. The significant improvement from the global affine transformations by allowing piecewise affine and thin-plate spline mappings confirmed our belief in the importance of non-rigid transformations. However, these results are clearly preliminary since the errors were



Selected Volume : None
Selected Surface : None
Selected Clip Plane : X1 (0.00x + 0.00y + 1.00z + 0.00 = 0)

Field of View:
Stereo Separation:
Slices: 256

Fig. 13. Brain mapping example. *Left*: the sulcal tracing tool with some traced sulci on the 3D MR brain volume. *Right*: Sulci extracted and displayed as point-sets.

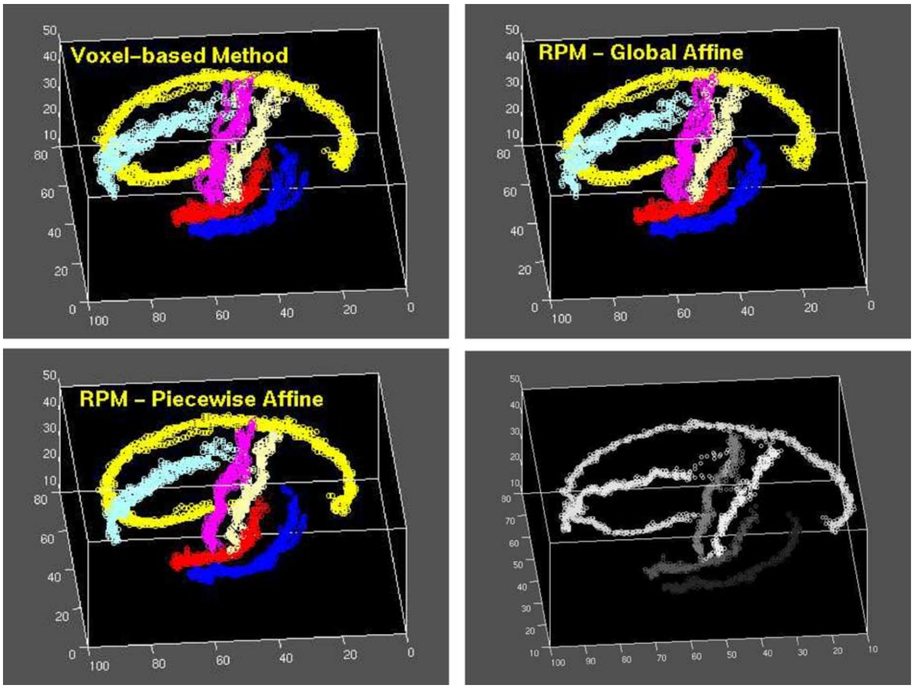


Fig. 14. Sulcal point mapping. The mapping results of five brains' sulcal point-sets on the left side of the brain are shown together. *Top left*: voxel-based mutual information with an affine mapping, *Top right*: RPM with an affine mapping, *Bottom left*: RPM with a piecewise affine mapping, *Bottom right*: TPS–RPM with a thin-plate spline mapping. Denser, closely packed distributions of sulcal points suggest that they are better aligned. We clearly see the improvement of RPM over the voxel-based approach at the sulci. This is especially obvious for TPS–RPM.

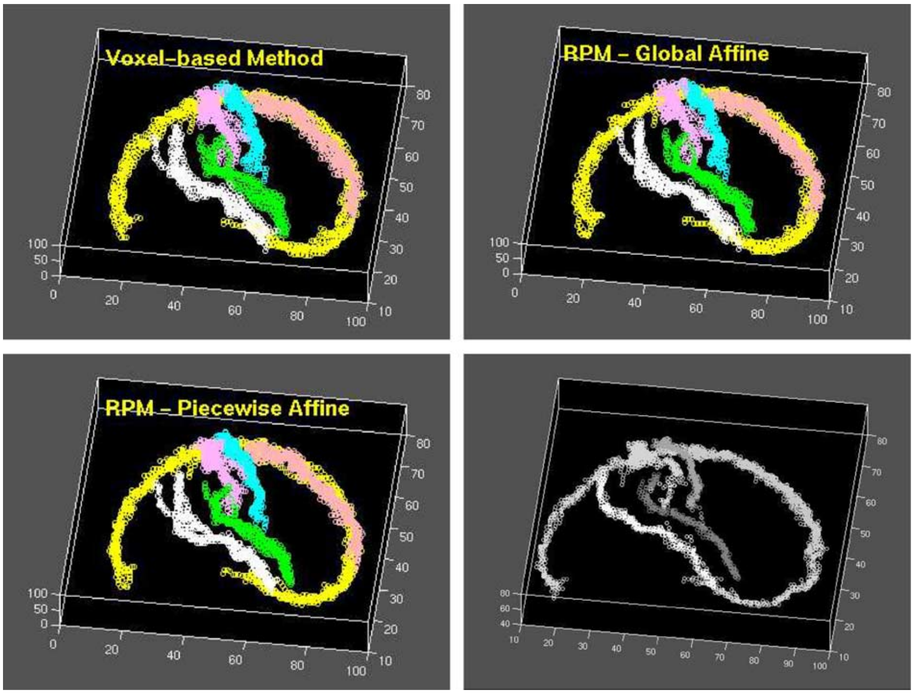


Fig. 15. Sulcal point mapping. Same as the above except the data from the other side of the brain are shown.

evaluated only at the sulci—exactly at the locations where a feature-based approach can be expected to perform best. Under no circumstances should this preliminary result be seen as demonstrating anything of clinical significance. Much more detailed validation and evaluation experiments comparing and contrasting voxel- and feature-based methods are required before any firm conclusions can be reached.

6. Discussion and conclusions

There are two important free parameters in the new non-rigid point matching algorithm—the regularization parameter λ and the outlier rejection parameter ζ. Below, we discuss ways of reducing the dependence on these free parameters.

Recall that the development of the TPS–RPM algorithm naturally lead to the regularization parameter value essentially being driven by the temperature. In the beginning, at high temperature, the regularization is very high and the algorithm focuses on rigid mappings. At lower temperatures, the regularization is much lower and non-rigid deformations emerge. This is no accident. When viewed from a mixture model perspective, the temperature can be interpreted as an isotropic variance parameter (on the matching likelihood). If we estimate the temperature parameter as a variance parameter, the somewhat artificial annealing schedule on λ can be removed. In addition, recall that we deal with fixed point-sets throughout. The transformation obtained at an intermediate stage is never used to generate a new point-set. If this is done, the dependence on λ is further reduced since the regularization parameter becomes a step-size parameter. This is exactly the approach taken in very recent and exciting work [8]. In [8], the authors clearly show how the TPS can be replaced by a diffeomorphism (using the TPS kernel) in landmark matching thereby reducing the role of the regularization parameter to that of a step-size parameter. Finally, the outlier rejection parameter ζ should also be viewed as related to the variance of the outlier bin in a mixture model and estimated. It should be mentioned that while the above gives us a principle for setting these free parameters, a parameter-free algorithm is usually notoriously difficult to achieve in practice.

We have developed a new non-rigid point matching algorithm—TPS–RPM—which is well suited for non-rigid registration. The algorithm utilizes the softassign, deterministic annealing, the thin-plate spline for the spatial mapping and outlier rejection to solve for both the correspondence and mapping parameters. The computational complexity of the algorithm is largely dependent on the implementation of the spline deformation [which can be $O(N^3)$ in the worst case]. We have conducted carefully designed synthetic experiments to empirically demonstrate the superiority of the TPS–RPM algorithm over TPS–ICP and have also applied the algorithm to perform non-rigid registration of cortical anatomical structures. While the algorithm is based on the notion of one-to-one correspondence, it is possible to extend it to the case of many-to-many matching [11] which is the case in *dense* (as opposed to sparse) feature-based registration. We plan to further explore [10] the applicability of point matching-based approaches to related problems in non-rigid registration such as anatomical atlases.

Acknowledgments

We acknowledge support from NSF IIS-0196457. We thank Larry Win, Jim Rambo, Bob Schultz and Jim Duncan for making available the sulcal data used in the brain mapping experiments and Colin Studholme for providing us with his implementation of UMDS. A reference MATLAB implementation of TPS–RPM (with many of the 2D examples used here) is available under the terms of the GNU General Public license (GPL) at http://www.cise.ufl.edu/~anand/students/chui/research.html.

References

[1] Y. Amit, A. Kong, Graphical templates for model recognition, IEEE Trans. Pattern Anal. Mach. Intell. 18 (4) (1996) 225–236.

[2] H. Baird, Model-based Image Matching Using Location, MIT Press, Cambridge, MA, 1984.

[3] D.H. Ballard, Generalized Hough transform to detect arbitrary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 13 (2) (1981) 111–122.

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

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

[6] R. Bhatia, Matrix analysis, in: Graduate Texts in Mathematics, 169, Springer, New York, 1996, p. 38.

[7] F.L. Bookstein, Principal warps: thin-plate splines and the decomposition of deformations, IEEE Trans. Pattern Anal. Mach. Intell. 11 (6) (1989) 567–585.

[8] V. Camion, L. Younes, Geodesic interpolating splines, in: Energy Minimization Methods for Computer Vision and Pattern Recognition (EMMCVPR), Springer, New York, 2001, pp. 513–527.

[9] H. Chui, J. Rambo, J. Duncan, R. Schultz, A. Rangarajan, Registration of cortical anatomical structures via robust 3Dpoint matching, in: Information Processing in Medical Imaging (IPMI), Springer, Berlin, 1999, pp. 168–181.

[10] H. Chui, A. Rangarajan, Learning an atlas from unlabeled point-sets, in: IEEE Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA), IEEE Press, New York, 2001, pp. 58–65.

[11] H. Chui, L. Win, J. Duncan, R. Schultz, A. Rangarajan, A unified non-rigid feature registration method for brain mapping. Med. Image Anal., in press, 2002.

[12] T. Cootes, C. Taylor, D. Cooper, J. Graham, Active shape models: their training and application, Comput. Vision Image Understanding 61 (1) (1995) 38–59.

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

[14] J. Feldmar, N. Ayache, Rigid, affine and locally affine registration of free-form surfaces, Internat. J. Comput. Vision 18 (2) (1996) 99–119.

[15] D. Geiger, F. Girosi, Parallel and deterministic algorithms from MRFs: surface reconstruction, IEEE Trans. Pattern Anal. Mach. Intell. 13 (5) (1991) 401–412.

[16] S. Gold, A. Rangarajan, A graduated assignment algorithm for graph matching, IEEE Trans. Pattern Anal. Mach. Intell. 18 (4) (1996) 377–388.

[17] S. Gold, A. Rangarajan, Softassign versus softmax: benchmarks in combinatorial optimization, in: D.S. Touretzky, M.C. Mozer, M.E. Hasselmo (Eds.), Advances in Neural Information Processing Systems (NIPS) 8, MIT Press, Cambridge, MA, 1996, pp. 626–632.

[18] S. Gold, A. Rangarajan, C.P. Lu, S. Pappu, E. Mjolsness, New algorithms for 2-D and 3-D point matching: pose estimation and correspondence, Pattern Recognition 31 (8) (1998) 1019–1031.

[19] E. Grimson, Object Recognition by Computer: The Role of Geometric Constraints, MIT Press, Cambridge, MA, 1990.

[20] E. Grimson, T. Lozano-Perez, Localizing overlapping parts by searching the interpretation tree, IEEE Trans. Pattern Anal. Mach. Intell. 9 (1987) 468–482.

[21] L.S. Hibbard, R.A. Hawkins, Objective image alignment for three-dimensional reconstruction of digital autoradiograms, J. Neurosci. Methods 26 (1988) 55–75.

[22] G. Hinton, C. Williams, M. Revow, Adaptive elastic models for hand-printed character recognition, in: J. Moody, S. Hanson, R. Lippmann (Eds.), Advances in Neural Information Processing Systems (NIPS) 4, Morgan Kaufmann, San Mateo, CA, 1992, pp. 512–519.

[23] T. Hofmann, J.M. Buhmann, Pairwise data clustering by deterministic annealing, IEEE Trans. Pattern Anal. Mach. Intell. 19 (1) (1997) 1–14.

[24] R. Hummel, H. Wolfson, Affine invariant matching, in: Proceedings of the DARPA Image Understanding Workshop, Cambridge, MA, 1988, pp. 351–364.

[25] D.P. Huttenlocher, G.A. Klanderman, W.J. Rucklidge, Comparing images using the Hausdorff distance, IEEE Trans. Pattern Anal. Mach. Intell. 15 (9) (1993) 850–863.

[26] Y. Lamdan, J. Schwartz, H. Wolfson, Object recognition by affine invariant matching, IEEE Conf. Comp. Vision, Pattern Recognition (1988) 335–344.

[27] G. Lohmann, D.Y. von Cramon, Automatic labelling of the human cortical surface using sulcal basins, Med. Image Anal. 4 (3) (2000) 179–188.

[28] D. Metaxas, E. Koh, N.I. Badler, Multi-level shape representation using global deformations and locally adaptive finite elements, Internat. J. Comput. Vision 25 (1) (1997) 49–61.

[29] C. Papadimitriou, K. Steiglitz, Combinatorial Optimization, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1982.

[30] J. Rambo, X. Zeng, R. Schultz, L. Win, L. Staib, J. Duncan, Platform for visualization and measurement of gray matter volume and surface area within discrete cortical regions from MR images, Neuroimage 7 (4) (1998) 795.

[31] A. Rangarajan, H. Chui, F. Bookstein, The softassign Procrustes matching algorithm, in: Information Processing in Medical Imaging (IPMI), Springer, Berlin, 1997, pp. 29–42.

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

[33] S. Sclaroff, A.P. Pentland, Modal matching for correspondence and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 17 (6) (1995) 545–561.

[34] G. Scott, C. Longuet-Higgins, An algorithm for associating the features of two images, Proc. R. Soc. Lond. B 244 (1991) 21–26.

[35] L. Shapiro, J. Brady, Feature-based correspondence: an eigenvector approach, Image Vision Comput. 10 (1992) 283–288.

[36] L.G. Shapiro, R.M. Haralick, Structural descriptions and inexact matching, IEEE Trans. Pattern Anal. Mach. Intell. 3 (9) (1981) 504–519.

[37] R. Sinkhorn, A relationship between arbitrary positive matrices and doubly stochastic matrices, Ann. Math. Statist. 35 (1964) 876–879.

[38] G. Stockman, Object recognition and localization via pose clustering, Comput. Vision Graph. Image Process. 40 (3) (1987) 361–387.

[39] C. Studholme, Measures of 3D medical image alignment, Ph.D. Thesis, University of London, London, UK, 1997.

[40] R. Szeliski, S. Lavallee, Matching 3D anatomical surfaces with non-rigid deformations using octree splines, Internat. J. Comput. Vision 18 (1996) 171–186.

[41] H. Tagare, D. O'Shea, A. Rangarajan, A geometric criterion for shape based non-rigid correspondence, in: Fifth Intl. Conf. Computer Vision (ICCV), 1995, pp. 434–439.

[42] S. Ullman, Aligning pictorial descriptions: an approach to object recognition, Cognition 32 (3) (1989) 193–254.

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

[44] W. Wells, Statistical approaches to feature-based object recognition, Internat. J. Comput. Vision 21 (1/2) (1997) 63–98.

[45] A.L. Yuille, Generalized deformable models statistical physics and matching problems, Neural Comput. 2 (1) (1990) 1–24.

[46] A.L. Yuille, N.M. Grzywacz, A mathematical analysis of the motion coherence theory, Internat. J. Comput. Vision 3 (2) (1989) 155–175.

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