# Robust Point Matching Revisited: A Concave Optimization Approach

Wei Lian[1] and Lei Zhang[2,⋆]

[1] Dept. of Computer Science, Changzhi University, Changzhi, 046011, Shanxi, China
lianwei3@gmail.com
[2] Dept. of Computing, The Hong Kong Polytechnic University
cslzhang@comp.polyu.edu.hk

**Abstract.** The well-known robust point matching (RPM) method uses deterministic annealing for optimization, and it has two problems. First, it cannot guarantee the global optimality of the solution and tends to align the centers of two point sets. Second, deformation needs to be regularized to avoid the generation of undesirable results. To address these problems, in this paper we first show that the energy function of RPM can be reduced to a concave function with very few non-rigid terms after eliminating the transformation variables and applying linear transformation; we then propose to use concave optimization technique to minimize the resulting energy function. The proposed method scales well with problem size, achieves the globally optimal solution, and does not need regularization for simple transformations such as similarity transform. Experiments on synthetic and real data validate the advantages of our method in comparison with state-of-the-art methods.

## 1 Introduction

Point matching is a fundamental yet challenging problem in computer vision, pattern recognition and medical image analysis. Many methods [1,2,3,4,5,6,7] have been proposed to solve the problem. Among them, the robust point matching (RPM) method [3] is very popular because of its robustness to many types of disturbances such as deformation, noise and outliers. Several variants [4,7,6] of RPM were later proposed.

In its basic form, RPM models point matching as a linear assignment−least square problem, where the energy function takes the following form:

$$E(P,\theta)=\sum_{i,j}p_{ij}\|y_j-T(x_i|\theta)\|^2+g(\theta) \tag{1}$$

Here $x_i$ and $y_j$ denote the model point $i$ and data point $j$, respectively. $P=\{p_{ij}\}$ denotes the correspondence matrix with $p_{ij}=1$ indicating that there is a correspondence between $x_i$ and $y_j$ and $p_{ij}=0$ otherwise. $T(\cdot|\theta)$ denotes the transformation with parameters $\theta$. $g$ denotes the regularization term used to

⋆ Corresponding author.

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part II, LNCS 7573, pp. 259–272, 2012.

regularize the value of $\theta$. To minimize (1), RPM relaxes $P$ to be continuously valued and employs deterministic annealing (DA) for optimization. Although RPM performs well in practice, it cannot guarantee the global optimality of the solution and tends to align the centers of two point sets. Besides, regularization $g$ is always needed to avoid the generation of undesirable results.

In this paper, we are interested in minimizing energy function (1) using global optimization techniques, instead of heuristic schemes such as DA. After eliminating the transformation variables, it can be observed that the minimization problem is reduced to a concave quadratic program, which facilitates the use of concave optimization techniques for global optimality. In contrast, RPM does not possess such property. Unfortunately, it is known that general concave optimization techniques are only suitable for small scale problems, whereas the number of variables in our problem is the product of the cardinalities of two point sets to be matched, which is quite large. Fortunately, after applying the eigen decomposition based linear transformation to our problem, the number of quadratic terms shrinks to be the number of transformation parameters, which is generally very small. It becomes a concave optimization problem where the number of variables in the linear part is much greater than the number of variables in the concave part. For this type of problems, there exist efficient optimization techniques [8]. Consequently, the proposed method scales well with problem size. Another advantage of our method over RPM lies in that it does not need regularization when simple transformations such as similarity transform is employed. In contrast, RPM always needs regularization to avoid the generation of undesirable results.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 discusses the proposed energy function. Section 4 presents the optimization algorithm. Section 5 presents the experimental results. Finally, Section 6 concludes the paper.

## 2 Related Work

**Point Matching.** The iterative closest point (ICP) method [1] iterates between finding point correspondence based on nearest neighbor relationship and updating transformation as a least square problem. However, ICP is not robust because of the discrete nature of point correspondence. To address the problem, RPM [3] relaxes point correspondence to be continuously valued and employs DA for optimization. Nonetheless, the optimization scheme of RPM is quite complex. To address the problem, the coherent point drift (CPD) method [4] models point matching under the probabilistic framework and uses the expectation-maximization (EM) technique for optimization. The methods in [5,6] view a point set as the result of sampling from an unknown distribution and convert point matching into the problem of matching corresponding distributions. The covariance driven correspondence method [7] uses the covariance of the transformation to guide the determination of point correspondences, but the method can only handle rigid transformations. The above methods are all heuristic schemes and therefore they can not guarantee the global optimality of the solution.

**Concave Optimization.** Concave optimization was used in [9] to solve the correspondence problems arising from computer vision, but the techniques employed there are only suitable for small scale problems. In contrast, our method scales well with problem size because of its special structure of optimization, which facilitates the use of large scale optimization techniques.

**Branch-and-Bound (B&B).** B&B is a popular technique for solving difficult nonlinear optimization problems. It was used in [10] to solve the rigid registration problem, where the correspondences between planes, lines or points are assumed to be known, and hence only the transformation needs to be solved, which involves a small number of variables. However, both point correspondence and transformation are not known in our problem, which makes the optimization much harder. In [11], two B&B methods were proposed to minimize the energy function of RPM. The first method is based on branching in the correspondence variable. But because of the high dimensionality of the correspondence variable, this method is only suitable for small scale problems. In contrast, our method works in the transformed variable space instead of the original correspondence variable space, and the number of variables to be branched equals the number of deformation parameters, which is generally very small. Therefore, our method has reasonable running time and scales well with problem size. The second method in [11] is based on branching in the transformation variable. Due to lack of properties such as concavity by the energy function, the lower bound of this method is not tight, and its convergence speed is slow.

## 3 Derivation of the Energy Function

Suppose that there are two point sets in $d-$dimensional space to be matched: the model point set $\mathscr{X} = \{x_i, i = 1, \ldots, m\}$, where point $x_i = \left[x_i^1, \cdots, x_i^d\right]^T$, and the data point set $\mathscr{Y} = \{y_j, j = 1, \ldots, n\}$, where point $y_j = \left[y_j^1, \cdots, y_j^d\right]^T$.

To make our problem tractable, we assumes that the transformation $T(x_i|\theta)$ is linear with respect to its parameters $\theta$, i.e., $T(x_i|\theta) = J(x_i)\theta$, where $J(x_i)$ is called the Jacobian matrix (examples include Eq. (13) and (14) in section 5). We consider the following form of regularization in this paper: $g(\theta) = (\theta-\theta_0)^T H(\theta-\theta_0)$, i.e., $\theta$ is required to be close to a constant vector $\theta_0$. Here $H$ is a positive semidefinite matrix whose elements represent the weights assigned to elements of $\theta$.

With the above consideration, the energy function (1) takes the following form:

$$E(P, \theta) = \sum_{i,j} p_{ij} \|y_j - J(x_i)\theta\|^2 + (\theta - \theta_0)^T H(\theta - \theta_0) \tag{2}$$

The correspondence matrix $P$ satisfies the two way normalization constraint:

$$P1_n = 1_m, \quad 1_m^T P \leq 1_n, \quad P \geq 0 \tag{3}$$

Here we require $P1_n = 1_m$, i.e., every point in $\mathscr{X}$ has a counterpart in $\mathscr{Y}$. This assumption is essential in simplifying formulas (see Eq. (4) and (5)) in the sequel. This assumption is commonly used in point matching literature [3].

$E$ is apparently a convex quadratic function of $\theta$. Therefore, by letting $\frac{\partial E}{\partial \theta} = 0$, we can get the optimal solution of $\theta$ as:

$$\hat{\theta} = (J^T J + H)^{-1} \left[ J^T (P \otimes I_d) Y + H\theta_0 \right] \tag{4}$$

where $J \triangleq \left[ J^T(x_1), \ldots, J^T(x_m) \right]^T$ and $Y \triangleq \left[ y_1^T, \ldots, y_n^T \right]^T$. Symbol $\otimes$ denotes the Kronecker product operator.

By substituting (4) into (2), the variable $\theta$ is eliminated and we arrive at an energy function only in $P$:

$$\begin{aligned} E(P) =& 1_m^T P Z - \left[ Y^T (P \otimes I_d)^T J + \theta_0^T H \right] (J^T J + H)^{-1} \\ & \cdot \left[ J^T (P \otimes I_d) Y + H\theta_0 \right] + \theta_0^T H \theta_0 \end{aligned} \tag{5}$$

where $Z \triangleq \left[ \|y_1\|_2^2, \ldots, \|y_n\|_2^2 \right]^T$. By eliminating the constant terms in (5), we get an equivalent optimization problem where the energy function is:

$$\begin{aligned} E'(P) =& 1_m^T P Z - 2\theta_0^T H (J^T J + H)^{-1} J^T (P \otimes I_d) Y \\ & - Y^T (P \otimes I_d)^T J (J^T J + H)^{-1} J^T (P \otimes I_d) Y \end{aligned} \tag{6}$$

Let the Cholesky factorization [12] of the positive definite matrix $(J^T J + H)^{-1}$ be

$$U^T U = (J^T J + H)^{-1} \tag{7}$$

where $U$ is an upper triangular matrix. Then we have

$$E'(P) = 1_m^T P Z - 2\theta_0^T H (J^T J + H)^{-1} J^T (P \otimes I_d) Y - \|U J^T (P \otimes I_d) Y\|_2^2 \tag{8}$$

It's apparent that $E'$ is a concave quadratic function of $P$. We have the following proposition.

**Proposition 1.** *There exists an optimal integer solution of $E'$.*

*Proof.* The constraint on $P$ in Eq. (3) satisfies the total unimodularity property [13], hence the vertices of the polytope (i.e., bounded polyhedron) determined by Eq. (3) will be integer valued. It's well known that the minimum solution of a concave function over a polytope can be obtained at its vertices. Therefore there exists an optimal integer solution of $E'$.

Based on the above Proposition 1, the output of our method will be binary point correspondences if simplex-like optimization algorithms are adopted, as in contrast to [14] where post-processing is needed to convert point correspondences to binary values. For the convenience of discussion in the sequel, we define the vectorization of matrix $P$ as the concatenation of its rows:

$$\mathrm{vec}(P) = [p_{11}, \ldots, p_{1n}, p_{21}, \ldots, p_{2n}, \ldots, p_{m1}, \ldots, p_{mn}]^T$$

Let $p = \mathrm{vec}(P)$, $E'$ becomes

$$\begin{aligned} E'(p) =& \left\{ 1_m^T \otimes Z^T - 2 \left[ \left( \theta_0^T H (J^T J + H)^{-1} J^T \right) \otimes Y^T \right] W \right\} p \\ & - \| \left[ (U J^T) \otimes Y^T \right] W p \|_2^2 \end{aligned} \tag{9}$$

Here the $mnd \times mn$ matrix $W \triangleq I_m \otimes \left[I_n \otimes e_d^1, \ldots, I_n \otimes e_d^d\right]^T$ satisfies $\mathrm{vec}(P \otimes I_d) = W\mathrm{vec}(P)$, where $e_d^i$ is a $d-$dimensional column vector with the $i$th element being 1 and the rest elements being 0s. $W$ is a large but sparse matrix and can be implemented using function *peye* in Matlab.

## 4 Optimization of the Energy Function

It is well known that a quadratic function can be converted into a separable form via linear transformation. It is also known that the convex envelope (i.e., the tightest convex underestimator) of a separable function over a rectangle is the sum of the convex envelope of each component of the function over the corresponding interval [8]. Based on these facts, the normal rectangular algorithm [8], which is specifically designed to minimize separable functions, is adopted in this paper to minimize the energy function (9). To this end, we first transform (9) into a separable form via eigen decomposition, and then derive the convex envelope of the resulting function over a rectangular region. We finally use the B&B technique [8] for optimization.

### 4.1 Eigen Decomposition

A quadratic function can be transformed into a separable function by a linear transformation of its variables, and the choice of the linear transformation is based on the eigen decomposition of the quadratic part of the function. Therefore, in this subsection, we focus on the eigen decomposition of the quadratic part of (9).

Let us define the $\#\theta \times mn$ matrix $A = \left[(UJ^T) \otimes Y^T\right] W$, where $\#\theta$ denotes the dimensionality of $\theta$. Then the quadratic part of (9) is $-p^T A^T A p$. Since $A^T A$ has dimension $mn \times mn$, which is very high, directly applying eigen decomposition to $A^T A$ is impractical. In the following, we propose an efficient way of finding the nonzero eigenvalues and eigenvectors of $A^T A$. Let the QR factorization of $A^T$ be:

$$QR = A^T$$

where $R$ is an upper triangular matrix. The columns of $Q$ are mutually orthogonal and have a unity norm.

It's apparent that

$$A^T A = QRR^T Q^T$$

Denote the eigenvalues and eigenvectors of $RR^T$ as $\lambda_i$ and $u_i^R$, $i = 1, \cdots, \#\theta$, respectively. Then the nonzero eigenvalues and eigenvectors of $A^T A$ are $\lambda_i$ and $u_i = Qu_i^R$, $i = 1, \cdots, \#\theta$, respectively. Therefore we can get the separable form of the quadratic part of (9) as:

$$-\|Ap\|_2^2 = -\sum_{i=1}^{\#\theta} \lambda_i (u_i^T p)^2 \tag{10}$$

### 4.2 Convex Envelope of the Energy Function over a Rectangle

With Eq. (10), $E'$ now takes the following separable form:

$$E'(p) = b^T p - \sum_{i=1}^{\#\theta} \lambda_i (u_i^T p)^2 \tag{11}$$

where $b \triangleq 1_m^T \otimes Z^T - 2\left[\left(\theta_0^T H (J^T J + H)^{-1} J^T\right) \otimes Y^T\right] W$.

For a separable function over a rectangle, its convex envelope can be readily obtained based on the following proposition [8]:

**Proposition 2.** *The convex envelope of a separable function* $\sum_{i=1}^{l} f_i(t_i)$ *over a rectangle* $M = \{t_i | r_i \leq t_i \leq s_i, i = 1 \ldots, l\}$ *equals the sum of the convex envelopes of components* $f_i(t_i)$ *over intervals* $[r_i, s_i], i = 1, \ldots, l$.

It's apparent that the convex envelope of $f(t) = -t^2$ over an interval $[r, s]$ is an affine function that agrees with $f$ at the endpoints of this interval: $f_M(t) = -(r+s)t + rs$, as illustrated in Fig. 1. Based on this fact and the above proposition, we can get the convex envelope of function (11) over a rectangle $M = \{p | r_i \leq u_i^T p \leq s_i, i = 1, \ldots, \#\theta\}$ as

$$E'_M(p) = b^T p - \sum_{i=1}^{\#\theta} \lambda_i (r_i + s_i) u_i^T p + \sum_{i=1}^{\#\theta} \lambda_i r_i s_i \tag{12}$$

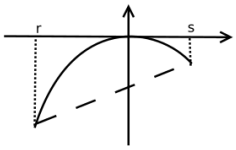


**Fig. 1.** The convex envelope (dashed line) of function $-t^2$ (solid line) over an interval $[r, s]$

### 4.3 Bisection of a Rectangle

We use the B&B algorithm [8] to find the global $\epsilon-$optimal solution of $E'$, i.e., a solution with function value no larger than $\epsilon$ from the global optimal value of $E'$. In the branching phase, a rectangle $M = \{p | r_i \leq u_i^T p \leq s_i, i = 1, \ldots, \#\theta\}$ is partitioned into two subrectangles. There are two issues to be addressed: 1) deciding the dimension along which to split the rectangle, and 2) along a chosen dimension where to split the rectangle. We use the bisection scheme [8] to solve these issues for its simplicity and effectiveness. For bisection, the second issue is addressed by choosing the midpoint as the splitting location, and the first issue is addressed based on the following fact: the difference between $f(t) = -t^2$ and its convex envelope $f_M(t) = -(r+s)t + rs$ over an interval $[r, s]$ satisfies:

$$\max\{f(t) - f_M(t), r \leq t \leq s\} = \frac{1}{4}(s - r)^2$$

Based on this fact, we can see that the dimension along which to split a rectangle should be chosen as $j \in \arg\max_i \frac{1}{4}\lambda_i(s_i - r_i)^2$. Given the optimal splitting dimension $j$, bisection results in two subrectangles: $M_1 = \{p \in M | u_j^T p \leq \frac{1}{2}(r_j + s_j)\}$ and $M_2 = \{p \in M | u_j^T p \geq \frac{1}{2}(r_j + s_j)\}$. It can be proved [8] that bisection leads to a B&B algorithm which is convergent.

### 4.4 Algorithm

We use the normal rectangular algorithm [8], a B&B approach specifically designed for separable functions, to optimize (11). During initialization, the bounding rectangle (i.e., the smallest rectangle containing the solution space) is computed. Then in each iteration of the algorithm, the rectangle yielding the lowest lower bound among all the rectangles is further subdivided so as to improve the global lower bound of the problem. Meanwhile, the upper bound is updated by evaluating the energy function with solutions of the linear programs used to compute the lower bound. The pseudo-code of the algorithm is summarized as follows.

**Initialization**

Select tolerance error $\epsilon > 0$.
Solve the $2\#\theta$ linear programs

$$\min_{p \in D} u_i^T p, \quad \max_{p \in D} u_i^T p$$

to obtain the basic optimal solutions $p^{0i}, \bar{p}^{0i}$ and the optimal values $\eta_i, \bar{\eta}_i$. Here $D$ denotes the solution space of $p$, as determined by Eq. (3). Clearly, $D \subset M_0 = \{p | \eta_i \leq u_i^T p \leq \bar{\eta}_i, i = 1, \ldots, \#\theta\}$. Set $\mathscr{M}_1 = \mathscr{N}_1 = \{M_0\}$, where $\mathscr{M}_1$ is the collection of all rectangles and $\mathscr{N}_1$ is the collection of active rectangles. Let $p^0 = \arg\min\{E'(p^{0i}), E'(\bar{p}^{0i}), i = 1, \ldots, \#\theta\}$.

**Iteration** $k = 1, 2, \ldots$

1. For each rectangle $M \in \mathscr{N}_k$, construct the convex envelope $E'_M(p)$ according to Eq. (12), and solve the linear program

$$\begin{aligned} &\min E'_M(p) \\ &s.t. \quad p \in D \cap M \end{aligned}$$

   to obtain a basic optimal solution $\omega(M)$ and the optimal value $\beta(M)$. $\beta(M)$ is the lower bound for region $D \cap M$.
2. Let $p^k$ equal to the best among all feasible solutions so far encountered: $p^{k-1}$ and all $\omega(M), M \in \mathscr{N}_k$. Delete all rectangles $M \in \mathscr{M}_k$ such that $\beta(M) \geq E'(p^k) - \epsilon$. Let $\mathscr{R}_k$ be the remaining collection of rectangles.
3. If $\mathscr{R}_k = \emptyset$, terminate: $p^k$ is the global $\epsilon-$minimal solution. Otherwise, go to Step 4.
4. Select the rectangle to be divided: $M_k = \{p | r_i^k \leq u_i^T p \leq s_i^k, i = 1, \ldots, \#\theta\} \in \arg\min\{\beta(M) | M \in \mathscr{R}_k\}$.

5. Let $i_k \in \arg\max_i\{\frac{1}{4}\lambda_i(s_i^k - r_i^k)^2\}$. Divide $M_k$ along dimension $i_k$ to get two subrectangles $M_{k1}$ and $M_{k2}$.
6. Let $\mathscr{N}_{k+1} = \{M_{k1}, M_{k2}\}$, $\mathscr{M}_{k+1} = (\mathscr{R}_k \backslash \{M_k\}) \cup \mathscr{N}_{k+1}$. Set $k \leftarrow k+1$ and return to Step 1.

Since the time complexity of the B&B algorithm is exponential in the worst case, the worst case time complexity of our algorithm is also exponential.

## 5 Experimental Results

We implement all the competing methods in Matlab R2010 on a PC with 2.4GHz CPU. Since all the competing methods output point correspondences, we use the correspondences computed by a method to find the best affine transformation between the two point sets, and define the error as the mean of the Euclidean distances between the affinely transformed model points and their ground truth data points. To get a complete picture of the performance of our method, in the following experiments we consider two cases for our method: $\theta$ is not regularized (i.e., $H = 0$) and $\theta$ is regularized. In the former case, our method becomes invariant to the corresponding transformation. The Matlab source code of this paper is available at: `http://www4.comp.polyu.edu.hk/~cslzhang/code.htm`.

### 5.1 Case One: $\theta$ Is Not Regularized

When $\theta$ is not regularized, with the increase of $\#\theta$, more degree of transformation freedom is allowed, and the matching results become less predictable. Our experimental results showed that for 2-D point matching, similarity transformation is a good trade-off between transformation flexibility and predictability of the matching results. For 2-D similarity transformation with parameters $\theta = \left[a, b, c, d\right]^T$, where $[c, d]^T$ is translation and $a = s\cos(\phi)$, $b = s\sin(\phi)$ with $s$ being scale and $\phi$ being rotation angle, the Jacobian matrix is

$$J(x_i) = \begin{bmatrix} x_i^1 & -x_i^2 & 1 & 0 \\ x_i^2 & x_i^1 & 0 & 1 \end{bmatrix} \tag{13}$$

Assuming that the data point set is unit sized, we set the tolerance error $\epsilon = 1$. Since the worst case complexity of our algorithm is exponential, we set the maximum search depth as 3 to make a good trade-off between running time and matching accuracy. We compare our method with 4 state-of-the-art methods: the unified graphical (UG) method [15] where similarity transformation is chosen, the fan-shaped triangulation (FST) method [16], the Viterbi algorithm (VA) based method [17] and the linear programming (LP) based method [14]. These methods can guarantee globally optimal or sub-optimal (for LP) solutions and some of them (UG and FST) are also rotation invariant, making them good candidates for comparison. For LP, we use shape context [18] as the feature descriptor. Since VA and LP are not rotation invariant, we render them rotation invariant by evaluating them on 8 evenly quantized angles and choosing the result with the minimum cost.

**Experiments on the Chui-Rangarajan Synthesized Data Sets.** Synthetic data are often used to quantitatively evaluate specific aspects of an algorithm. In this subsection, we use the Chui-Rangarajan synthesized data sets [3] to test the 5 competing methods' robustness against non-rigid deformation, noise in position and outliers. In each test, the model shape is subject to first random rotation and then one of the above distortions to generate a data point set. The model shapes (a tropical fish and a Chinese character) and examples of data point sets in the 3 categories of tests are shown in Fig. 2.

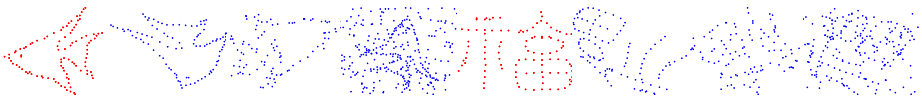

**Fig. 2.** For every 4 columns, from left to right: the model point set and examples of data point set in the deformation, noise and outlier tests, respectively

The matching errors of the methods are shown in Fig. 3. It can be seen that for the deformation and noise tests, our method outperforms other methods, demonstrating its robustness to deformation and noise. For the outlier test, our method performs in average compared with other methods for the fish test while slightly worse than other methods for the Chinese character test. Part of the reason is that all the other methods except for UG employ the shape context feature descriptor for matching, which can much increase their robustness to

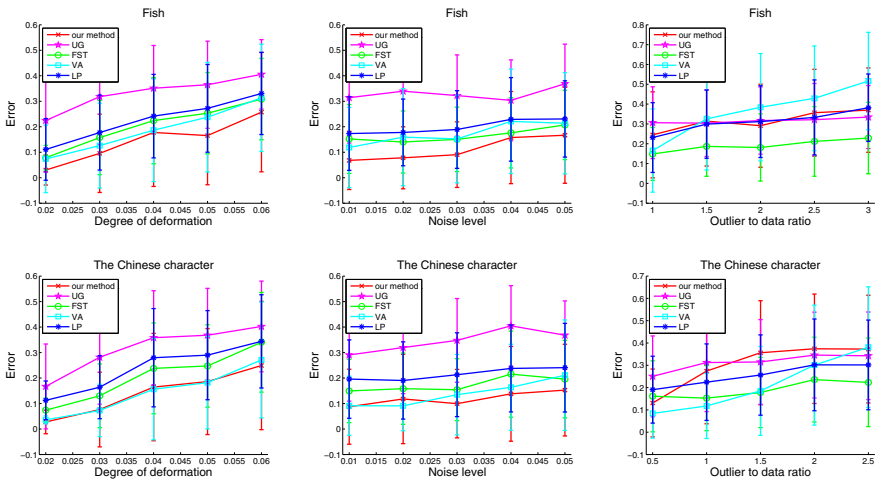


**Fig. 3.** Matching errors by the 5 competing methods on the Chui-Rangarajan synthesized data sets. The error bars indicate the standard deviation of the error over 100 random trials.

outliers. In comparison, our method and UG only use the information of point positions for matching.

The average running time of the 5 competing methods are listed in Table 1. The proposed method is the slowest among them. This is because linear programming needs to be performed at each iteration of our method. Nonetheless, the high running time is the price for the global optimality of the proposed scheme. It can also be seen that the running time of our method is acceptable when the number of points is not big. When the number of points becomes large (e.g., in the case of outliers), the running time of our method increases much.

**Table 1.** Average running time (in seconds)

| | Deformation | Noise | Outliers |
|---|---|---|---|
| our method | 23.4567 | 23.9379 | 162.6343 |
| UG | 10.6118 | 10.4448 | 157.1455 |
| FST | 7.8901 | 8.8471 | 46.3078 |
| VA | 4.0145 | 4.0323 | 14.3741 |
| LP | 19.5272 | 18.4779 | 31.3505 |

**Experiments on Image Sequences.** We then test the 5 competing methods by using the CMU hotel and house sequences (CMU Image Database: `http://vasc.ri.cmu.edu/idb/html/motion/`) and two other image sequences from [19]. Similar to the experimental set up in [20], for each image sequence, we select 30 feature points and manually track them over the image sequence to generate a sequence of point sets. For two images separated by different sequence gap, the associated pair of point sets are used for matching, as illustrated in the 1st and 3rd columns of Fig. 4 and Fig. 5, respectively. The matching accuracy (fraction of correct correspondences) by the competing methods are shown in the 2nd and 4th columns of Fig. 4 and Fig. 5, respectively. It can be seen that our method has higher matching accuracy than other methods. In particular, it achieves almost 100% accuracy on the CMU sequence tests over a wide range

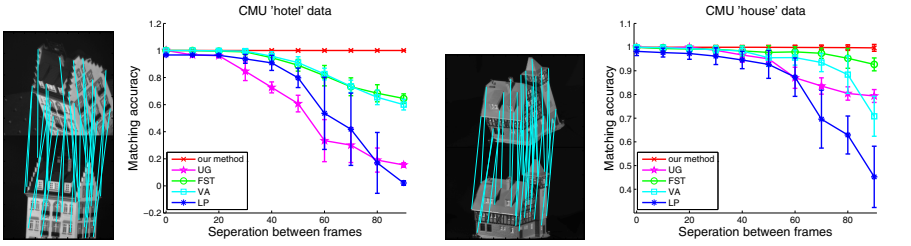


**Fig. 4.** 1st and 3rd columns: examples of input image pair and ground truth point correspondences; 2nd and 4th columns: average matching accuracy by the 5 competing methods. The error bars indicate the standard deviation of the error.

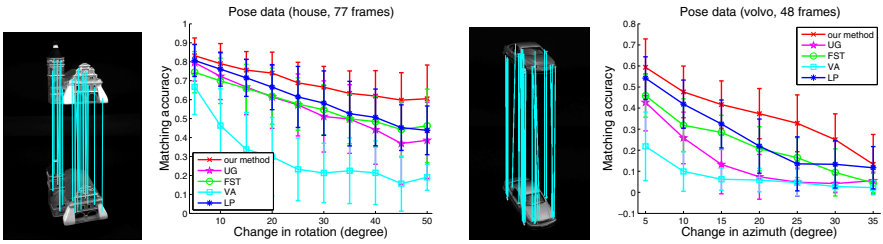


**Fig. 5.** 1st and 3rd columns: examples of input image pair and ground truth point correspondences; 2nd and 4th columns: average matching accuracy by the 5 competing methods. The error bars indicate the standard deviation of the error.

of frame intervals. For the other 2 sequences, our method outperforms other methods by a large margin (see Fig. 5).

### 5.2 Case Two: $\theta$ is Regularized

When $\theta$ is regularized, our method becomes transformation variant, but the benefit is that our method's matching accuracy and robustness to disturbances can be improved.

We use affine transformation in our method as it is the simplest non-rigid transformation and the number of parameters is small (note that our method's running time depends on the number of transformation parameters). For 2-D affine transformation with parameters $\theta = \left[a, b, c, d, e, f\right]^T$, where $[a, b, c, d]^T$ is the linear part of the transformation and $[e, f]^T$ is translation, the Jacobian matrix is

$$J(x_i) = \begin{bmatrix} x_i^1 & x_i^2 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i^1 & x_i^2 & 0 & 1 \end{bmatrix} \tag{14}$$

We choose the weighting matrix as $H = diag([1\ 1\ 1\ 1\ 0\ 0])$, i.e., we only regularize the linear part of the transformation. We choose $\theta_0 = [1\ 0\ 0\ 1\ 0\ 0]^T$, i.e., the linear part of the transformation should be close to the identity transformation (please refer to Eq. (2)).

The tolerance error is set as $\epsilon = 0.06m$, i.e., proportional to the number of model points. We compare our method with 3 methods: RPM [3], CPD [4] and the local neighborhood structure preserving (LNSP) method [21]. These 3 methods are very popular and represent state-of-the-art. Affine transformation is used for RPM and CPD.

First, we use the Chui-Rangarajan synthesized data sets to test the 4 competing methods' robustness against non-rigid deformation and noise (outlier is not tested; instead, it is replaced by a more challenging clutter test in the following). The experimental set up is similar to that in section 5.1 except that there is no rotation between two point sets. Performances of the 4 methods are shown in Fig. 6. It can be seen that our method has the lowest error among all the methods.

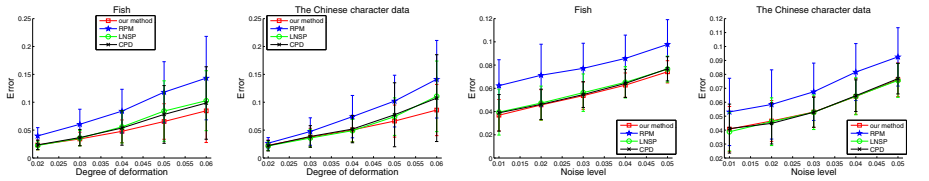


**Fig. 6.** Matching errors by the 4 competing methods on the Chui-Rangarajan synthesized data sets. The error bars indicate the standard deviation of the error over 100 random trials.

Second, we test the robustness of the 4 methods against clutter. The experimental set up is as follows. For a shape which is represented as a point set, we first obtain its shortest Hamiltonian cycle by solving a traveling salesman problem. Then a segment of the cycle starting at a random point and with different length is chosen as the model point set (the remaining points correspond to clutter). The moderately non-rigidly deformed version of the original shape is chosen as the data point set. The two shapes, the tropical fish and the Chinese character, as shown in the 1st and 5th columns of Fig. 2, are used in the experiments. Examples of model and data point sets are shown in the top row of Fig. 7.

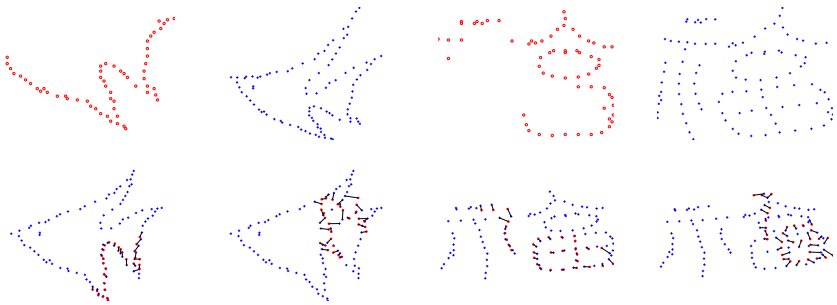

**Fig. 7.** Top row: examples of model (1st and 3rd columns) and data (2nd and 4th columns) point sets used in the clutter tests. Bottom row: examples of matching results by our method (1st and 3rd columns) and RPM (2nd and 4th columns), where the affinely transformed model points are shown as red $*$ and point correspondences are indicated by black line segments.

Performances of the 4 methods are shown in Fig. 8. It can be seen that our method is less sensitive to clutter and its error keeps almost unchanged when clutter becomes severe. In contrast, for other methods the errors increase quickly with the increase of the severity of clutter. This demonstrates the robustness of our method against clutter. Examples of matching results by our method and RPM are shown in the bottom row of Fig. 7.

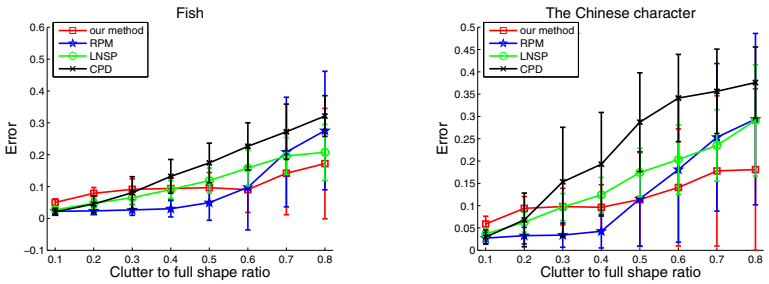


**Fig. 8.** Matching errors by the 4 competing methods on the test of clutter. The error bars indicate the standard deviation of the error over 100 random trials.

## 6 Conclusion

We proposed a new approach to minimizing the energy function of the classical RPM method. After eliminating the transformation variable, we reduced the energy function to a concave quadratic program, which can be efficiently solved by large scale concave optimization techniques. Our method can guarantee the global optimality of the solution, and does not need to regularize deformation for simple transformations such as similarity transform. Our method also scales well with problem size due to the special structure of its optimization problem. Extensive experimental results demonstrated that the proposed method has high matching accuracy and high robustness to disturbances, such as clutter, in comparison with state-of-the-art methods.

**Acknowledgement.** This work is supported by the Natural Science Foundation for Young Scientists of Shanxi Province, China (Grant No. 2012021015-2).

## References

1. Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. IEEE Trans. Pattern Analysis and Machine Intelligence 14, 239–256 (1992)
2. Zhang, Z.: Iterative point matching for registration of free-form curves and surfaces. International Journal of Computer Vision 13, 119–152 (1994)
3. Chui, H., Rangarajan, A.: A new point matching algorithm for non-rigid registration. Computer Vision and Image Understanding 89, 114–141 (2003)
4. Myronenko, A., Song, X.: Point set registration: Coherent point drift. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 2262–2275 (2010)
5. Tsin, Y., Kanade, T.: A Correlation-Based Approach to Robust Point Set Registration. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3023, pp. 558–569. Springer, Heidelberg (2004)
6. Jian, B., Vemuri, B.C.: A robust algorithm for point set registration using mixture of gaussians. In: IEEE International Conference on Computer Vision, vol. 2, pp. 1246–1251 (2005)

7. Sofka, M., Yang, G., Stewart, C.V.: Simultaneous covariance driven correspondence (cdc) and transformation estimation in the expectation maximization framework. In: IEEE Conf. Computer Vision and Pattern Recognition, pp. 1–8 (2007)
8. Horst, R., Tuy, H.: Global Optimization, Deterministic Approaches. Springer (1996)
9. Maciel, J., Costeira, J.: A global solution to sparse correspondence problems. IEEE Trans. Pattern Analysis and Machine Intelligence 25, 187–199 (2003)
10. Olsson, C., Kahl, F., Oskarsson, M.: Branch-and-bound methods for euclidean registration problems. IEEE Transactions on Pattern Analysis and Machine Intelligence 31, 783–794 (2009)
11. Pfeuffer, F., Stiglmayr, M., Klamroth, K.: Discrete and geometric branch and bound algorithms for medical image registration. Annals of Operations Research 196, 737–765 (2012)
12. Golub, G.H., Loan, C.F.V.: Matrix computations. The Johns Hopkins University Press (1996)
13. Papadimitriou, C.H., Steiglitz, K.: Combinatorial optimization: algorithms and complexity. Dover Publications, Inc., New York (1998)
14. Jiang, H., Drew, M.S., Li, Z.-N.: Matching by linear programming and successive convexification. IEEE Trans. Pattern Analysis and Machine Intelligence 29, 959–975 (2007)
15. Caetano, T.S., Caelli, T.: A unified formulation of invariant point pattern matching. In: 18th International Conference on Pattern Recognition (ICPR 2006), vol. 3, pp. 121–124 (2006)
16. Lian, W., Zhang, L.: Rotation Invariant Non-rigid Shape Matching in Cluttered Scenes. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 506–518. Springer, Heidelberg (2010)
17. Thayananthan, A., Stenger, B., Torr, P.H.S., Cipolla, R.: Shape context and chamfer matching in cluttered scenes. In: IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 127–133 (2003)
18. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Analysis and Machine Intelligence 24(4), 509–522 (2002)
19. Vikstén, F., Forssén, P.E., Johansson, B., Moe, A.: Comparison of local image descriptors for full 6 degree-of-freedom pose estimation. In: IEEE International Conference on Robotics and Automation, pp. 2779–2786 (2009)
20. Caetano, T.S., Caelli, T., Schuurmans, D., Barone, D.A.C.: Graphical models and point pattern matching. IEEE Trans. Pattern Analysis and Machine Intelligence 28(10), 1646–1663 (2006)
21. Zheng, Y., Doermann, D.: Robust point matching for nonrigid shapes by preserving local neighborhood structures. IEEE Trans. Pattern Analysis and Machine Intelligence 28, 643–649 (2006)