# Rotation Invariant Nonrigid Point Set Matching in Cluttered Scenes

Wei Lian, Lei Zhang, *Member, IEEE*, and David Zhang, *Fellow, IEEE*

***Abstract*—This paper addresses the problem of rotation invariant nonrigid point set matching. The shape context (SC) feature descriptor is used because of its strong discriminative nature, while edges in the graphs constructed by point sets are used to determine the orientations of SCs. Like lengths or directions, oriented SCs constructed in this way can be regarded as attributes of edges. By matching edges between two point sets, rotation invariance is achieved. Two novel ways of constructing graphs on model point set are proposed, aiming at making the orientations of SCs as robust to disturbances as possible. The structures of these graphs facilitate the use of dynamic programming (DP) for optimization. The strong discriminative nature of SC, the special structure of the model graphs, and the global optimality of DP make our methods robust to various types of disturbances, particularly clutters. The extensive experiments on both synthetic and real data validated the robustness of the proposed methods to various types of disturbances. They can robustly detect the desired shapes in complex and highly cluttered scenes.**

***Index Terms*—dynamic programming, shape context, shape representation, point set matching**

## I. Introduction

In many applications of computer vision, pattern recognition and medical image analysis, one common procedure is to match two or more point sets, and nonrigid point set matching is particularly difficult because the possible nonrigid deformation of the model shape is numerous [1]. In practice, the scene is often contaminated by clutters, making the point set matching problem more complicated. In this paper, we focus on how to locate a deformable shape in cluttered scenes under the nonrigid point set matching framework. The shape may undergo arbitrary translational and rotational changes, and it may be nonrigidly deformed and corrupted by clutters.

Various approaches have been proposed to solve the many difficult problems in point set matching [2]. Shape context (SC) [3] is a widely used feature descriptor for point set matching. The SC of a point measures the distribution of the relative positions of its neighbors. Denote by $h_i(k)$ and $h'_j(k)$, $k = 1, \dots, K$, the SCs at point $i$ in the model point set and point $j$ in the data point set, respectively. In [3], the $\chi^2$ test statistic was used to measure their distance:

$$\frac{1}{2}\sum_{k=1}^{K}\frac{[h_i(k)-h'_j(k)]^2}{h_i(k)+h'_j(k)} \tag{1}$$

Wei Lian is with the Department of Computer Science, Changzhi University, Changzhi, 046011, Shanxi, P.R. China. Lei Zhang and David Zhang are with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. Corresponding author: Lei Zhang, Email: *cslzhang@comp.polyu.edu.hk*. This work is supported by the Hong Kong SAR General Research Fund (PolyU 5330/07E).

SC is very discriminative and quite robust to various types of disturbances, which makes it a useful tool for nonrigid point set matching. However, unlike image feature descriptors such as SIFT [4], [5], GLOH [6] and ZM phase [7], which are rotation invariant, it is difficult to reliably enable SC rotation invariant because point sets contain much less redundant information than images. Prior attempts at making SC rotation invariant are not very successful. For example, in [3], tangent directions are used to determine the orientations of SCs, but they are susceptible to positional noise. In [8], the distance between two SCs is rendered rotation invariant by trying exhaustively all relative rotations between them, computing the corresponding distances and then selecting the minimum distance. However, this can significantly degrade the discriminative power of SC. In [9], the mass center of a point set is used as the reference position to determine the orientations of SCs, but this method requires that two point sets' mass centers must roughly correspond to each other.

Because of the strong discriminative power of SC, it is of great interest to investigate the use of SC in rotation invariant nonrigid point set matching. To this end, we propose to use edges in the graph constructed by a point set to determine the orientations of SCs. Like lengths or directions, oriented SCs constructed in this way can be regarded as the attributes of edges. By edge matching between two point sets, rotation invariance can be achieved. Clearly, the point set matching problem is now converted to a graph matching problem, and one key issue is how to construct graphs on the two point sets. To make the problem tractable, we assume that the model point set can be embedded in the data point set, i.e., each model point can find a counterpart in the data points. With this assumption, an edge in the model graph is likely to be matched to any pair of points in the data point set. Therefore, the most appropriate graph for the data point set should be the complete graph [10] where any two points are adjacent. For the model point set, two novel approaches to constructing graphs will be proposed in this paper: minimum spanning tree (MST) induced triangulation and star graph [11].

For MST induced triangulation, there are two types of edges: the "frame" edges formed by choosing one point as the reference and connecting it to the rest points, and the "boundary" edges generated in the MST constructed by the model points except for the reference point. The graph corresponding to MST induced triangulation is a 2-tree [10], whose best embedding in the data point set can be found by using dynamic programming (DP). Nonetheless, the time complexity of such a scheme is relatively high. Star graph is then proposed as an alternative for graph construction, and it involves only the frame edges generated in MST induced triangulation. Edges in a star graph form a tree so that DP can be used to find the best embedding of the graph in the data point set but the time complexity is much lower than that of MST induced triangulation.

Compared with previous attempts at enabling SC rotation invariant, the proposed methods retain the discriminative power of SC while being robust to orientation disturbances. MST induced triangulation can be viewed as an improvement over our previously proposed fan-shaped triangulation scheme [12] in order to better handle the case when shapes cannot be easily represented as simple polygons. In such cases, MST is better than simple polygons to represent shapes. The proposed MST induced triangulation based matching also shares similarities with the method in [13] in that triangles are used to represent shapes and DP is used to find the best embedding of triangles in data set. However, the method in [13] is for deformable template detection in images. Triangulation was used to introduce nonrigid deformation in template, and the constrained Delaunay triangulation [14] was used to maximize the effect. In contrast, the purpose of triangulation in our method is to enable SC rotation invariant, and a different triangulation scheme is adopted to ensure the orientation of SC being as robust to disturbances as possible.

The rest of the paper is organized as follows. Section II reviews some related work. Section III and Section IV present the shape representations based on MST induced triangulation and star graph, and their applications in point set matching, respectively. Section V presents extensive experiments and Section VI concludes the paper.

## II. Related work

There are mainly two categories of variables in point set matching: point correspondence and spatial mapping. As the methods proposed in this paper only need to model point correspondence, we will review those previous work where only point correspondence is concerned.

When only point correspondence is concerned, point set matching can be formulated as a graph matching problem. State-of-the-art graph matching methods include graduated assignment [15], spectral methods [16]–[18] and semidefinite relaxation [19]. Theory of dual decomposition was used to combine several optimization techniques to solve the graph matching problem in [20]. Point set matching was formulated as an embedding problem in [21], where the to-be-registered point sets were embedded in the same Euclidean space by constructing a graph on them. The edges within individual point set were constructed based on spatial arrangement and the edges between different point sets were constructed based on feature similarity. The embedding was then implemented by solving an Eigen-value problem. In [22], the geometric neighborhood (i.e., graph) of point correspondences was represented as the categorical graph product of two point sets' spatial graphs. The graph Laplacian was then used to regularize point correspondence during optimization. In [9], point set matching was formulated as the matching of neighborhood graphs (there is an edge between two points if they are neighbors) of two point sets. Relaxation labeling was used for optimization. This method exhibits good robustness to nonrigid deformation and positional noise, but it is not robust to outliers. Besides, to enable rotation invariance, it requires that the two point sets' mass centers should correspond to each other. For problems involving missing or erroneous structures, this requirement is difficult to satisfy. Our proposed methods can be interpreted as graph matching approaches as well, but they do not require that two point sets' mass centers must correspond to each other.

Linear programming is a commonly used optimization technique in computer vision. In [23], it was used to minimize the feature matching cost and preserve both the shape and orientation of the model point set. The lower convex hull property was used to speed up the method, which makes the complexity of the method essentially independent of the number of data points. The method was extended to be similarity invariant in [8] by explicitly modeling the rotation and scaling between two point sets, and was extended to be affine invariant in [24] by using a technique similar to local linear embedding [25]. In both the extensions, the distance between two comparing features is required to be the same when the two features are relatively rotated. For rotation variant features, such as SC, the problem is solved by trying all relative rotations between the two features, computing the corresponding distances and then selecting the minimum distance. However, this can significantly degrade the discriminative power of features. Fortunately, the proposed methods can circumvent this problem by converting point matching to a graph matching problem, where edges are used to determine the orientations of features without requiring the features being rotation invariant.

DP is an optimization technique commonly used for matching chains or trees [26], [27]. Due to the global optimality and discrete nature of DP, matching methods based on DP are very robust to clutters. DP can also be used to match triangulated (i.e., chordal) graphs [28] where the perfect elimination scheme [10] of the vertices can be used to design recursive equations used by DP. For example, in [29], $k$-tree (a subclass of chordal graph) was used to preserve rigidity of a point set. The method was later extended to be similarity, affine or projective invariant in [30]. Constrained Delaunay triangulation was used to introduce deformation into template in [13]. The complexity of DP depends exponentially on the size of the maximal clique of a chordal graph, which restricts the applicability of DP to graphs with large clique size. To address this problem, speedup measures [31], [32] have been proposed for the special case that the energy function only contains unary and pairwise terms.

## III. Shape representation based on MST induced triangulation for point set matching

In our previous work [12], fan-shaped triangulation was proposed for shape representation to enable SC rotation invariant. It assumes that the model point set resembles the shape of a simple polygon, which is obtained via finding the shortest Hamiltonian cycle [33]. However, not all shapes can be easily represented as simple polygons (e.g., the Chinese characters). Compared with the shortest Hamiltonian cycle, MST is a better candidate for shape representation. When a point set has the shape of a simple polygon, the shape of MST will be close to the shortest Hamiltonian cycle (please refer to the first row of Fig. 1). When a point set can not be easily represented as a simple polygon, MST will work better at representing the shape of a point set than the shortest Hamiltonian cycle (please refer to the second row of Fig. 1).

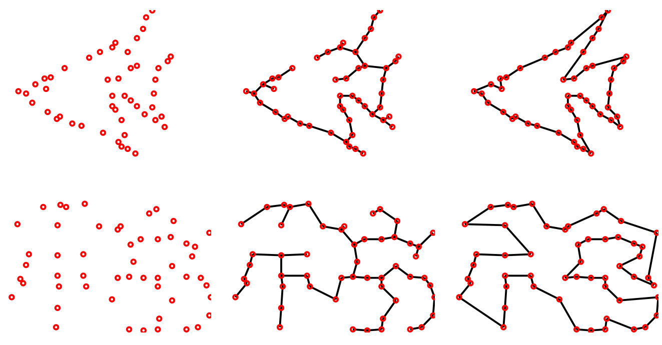

Fig. 1. Comparison of MST and the shortest Hamiltonian cycle at representing the shapes of point sets. Left column: point sets. Middle column: the MSTs of the point sets. Right column: the shortest Hamiltonian cycles of the point sets.

### A. MST induced triangulation

Let's denote the 2-D model point set as $\mathscr{X} = \{x_i, 0 \leq i \leq n\}$, where $x_i$ is the coordinate of the $i$th point. Denote by $v_i$ the vertex corresponding to the $i$th point. The MST induced triangulation of $\mathscr{X}$ consists of the following steps. 1) A point is chosen as the reference such that the average distance from it to the rest points is maximized. Without loss of generality, let's take $x_0$ as the reference point. We connect $x_0$ to points in $\mathscr{X}\backslash\{x_0\}$ with $n$ edges (which are called "frame" edges). 2) We construct MST on $\mathscr{X}\backslash\{x_0\}$ (the resulting graph is denoted as $\text{MST}_{\mathscr{X}\backslash\{x_0\}}$ and we call the edges in $\text{MST}_{\mathscr{X}\backslash\{x_0\}}$ as "boundary" edges). This leads to a triangulation of $\mathscr{X}$, where each triangle consists of two frame edges and one boundary edge. Fig. 2 illustrates the triangulation process. The graph constructed in this way is denoted as $G_{\mathscr{X}}$. For $G_{\mathscr{X}}$, the deformations of individual triangles will aggregate to form the overall nonrigid deformation of $\mathscr{X}$. This mechanism is similar to that described in [13]. Therefore our method is capable of handling nonrigid deformation to a certain extent.

It can be proved that $G_{\mathscr{X}}$ is a 2-tree. Before giving the proof, let's review some basic concepts in graph theory. A $k$-clique is a set of $k$ vertices where any two vertices are adjacent. A 2-clique is simply an edge. Let $G = (V, E)$ be a graph, where $V$ is the set of all vertices and $E$ is the set of all edges. For a subset $S \subset V$, the graph *induced* by $S$ is defined as $S$ itself and all the edges in $E$ having both endpoints in $S$.

*Definition 1:* A graph is called a $k$-tree if there is an ordering $\sigma = (u_1, \ldots, u_n)$ of its vertices such that for the subgraph induced by $S = \{u_i, \ldots, u_n\}$, $1 \leq i \leq n - k$, the neighboring vertices of $u_i$ form a $k$-clique. The ordering $\sigma$ is called a perfect elimination scheme for the vertices of the graph.

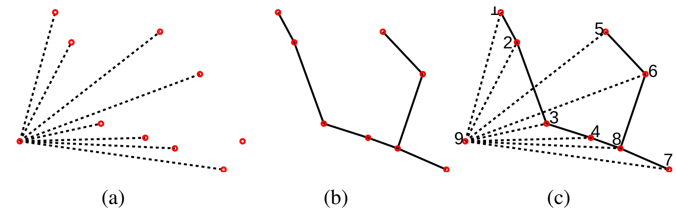


Fig. 2. Steps of MST induced triangulation. (a) Frame edges (dashed line segments) formed by connecting the reference point to all the other points. (b) MST (solid line segments) of all the points except for the reference point. (c) Triangulation result by combining both types of edges. The perfect elimination scheme is indicated by the numbers associated to vertices.

*Proposition 1:* $G_{\mathscr{X}}$ is a 2-tree.

*Proof:* Since $MST_{\mathscr{X}\backslash\{x_0\}}$ is a tree, we can get a perfect elimination scheme for its vertices by deleting its leaves one by one. Without loss of generality, let's assume that this perfect elimination scheme is $(v_1, \ldots, v_n)$. Then it can be verified that $(v_1, \ldots, v_n, v_0)$ is a perfect elimination scheme for vertices in $G_{\mathscr{X}}$. For the subgraph induced by $S = \{v_i, \ldots, v_n, v_0\}$, $1 \leq i \leq n - 1$, the neighboring vertices of $v_i$ form an edge (with one of the endpoints being $v_0$), as illustrated in Fig. 2 (c). ■

### B. Oriented SC features

We then compute oriented SC [3] for each point in $\mathscr{X}\backslash\{x_0\}$, whose positive x-axis is directed at $x_0$, coinciding with the direction of the corresponding frame edge, as illustrated in Fig. 3. This operation has time complexity $O(n)$ and space complexity $O(n)$. Similar to lengths or directions, oriented SC features constructed in this way can be regarded as attributes of frame edges. By matching edges between two point sets, we can achieve the goal of enabling SC rotation invariant. Due to the strong discrimination power of SC, the robustness of our method to various types of disturbances is greatly enhanced.

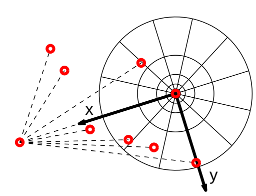


Fig. 3. Construction of oriented SC. The x-axis of SC for a point is directed at the reference point. Frame edges are shown as dashed line segments.

The reason we use a single reference point to determine the orientation of SC and select the reference point such that the average distance from it to the rest points is maximized is based on the following observation. First, directions of longer edges are less affected by positional jitter of their endpoints. More specifically, for an edge $(x_1, x_2)$ with endpoints $x_i = \bar{x}_i + \Delta x_i, i = 1, 2$, where $\bar{x}_i$ is the noise free position and $\Delta x_i$ is noise, the direction of the edge is

$$\frac{x_2 - x_1}{\|x_2 - x_1\|} \approx \frac{x_2 - x_1}{\|\bar{x}_2 - \bar{x}_1\|} = \frac{\bar{x}_2 - \bar{x}_1}{\|\bar{x}_2 - \bar{x}_1\|} + \frac{\Delta x_2 - \Delta x_1}{\|\bar{x}_2 - \bar{x}_1\|} \quad (2)$$

Here the second term is contributed by noise. Therefore the larger the length $\|\bar{x}_2 - \bar{x}_1\|$ is, the less influence the noise will impose on the direction of the edge.

Second, using a single reference point is better than using multiple reference points for the robustness of edge directions. Specifically, for two edges $(x_1, x_2)$ and $(x_1, x_3)$, their relative angle will be affected by positional jitter of the 3 endpoints $x_i, i = 1, 2, 3$. While for two edges $(x_1, x_2)$ and $(x_3, x_4)$, their relative angle will be affected by positional jitter of the 4 endpoints $x_i, i = 1, \dots, 4$, which is less robust than that by using 3 endpoints $x_i, i = 1, 2, 3$. Therefore using a single reference point is more favorable for robustness of edge directions.

### C. Shape representation for data point set

Let's denote the 2-D data point set as $\mathscr{Y} = \{y_j, 0 \leq j \leq m\}$, where $y_j$ is the coordinate of the $j$th point. Since an edge in $\mathscr{X}$ can be matched to any pair of points in $\mathscr{Y}$, we choose a complete graph from $\mathscr{Y}$ for matching. For each edge, we compute the oriented SC for each endpoint of it with the positive x-axis directed at the other endpoint of the edge. This process has time complexity $O(m^2)$ and space complexity $O(m^2)$.

In practice, the time of computing SC features for a point in $\mathscr{Y}$ with all the rest points serving as possible references can be reduced in the following way: for this point, the SC features with the positive x-axis oriented respectively to angles $0, \frac{1}{M}2\pi, \dots, \frac{M-1}{M}2\pi$ (we set $M = 50$ in our algorithm) are constructed. The SC features with all the rest points being possible references are substituted by these SC features based on orientation proximity. With this heuristic, the time complexity of computing SC features in $\mathscr{Y}$ is essentially $O(mM)$.

### D. Energy function

Given $\mathscr{X}$ and $\mathscr{Y}$, the task of matching is to find a mapping $\phi : \mathscr{X} \to \mathscr{Y}$ which maps the $i$th point in $\mathscr{X}$ to the $l_i$th point in $\mathscr{Y}$ so that certain energy function can be minimized. Our energy function takes the following form:

$$E(\phi) = \sum_{i=1}^{n} D_{sc}[i,0](l_i, l_0) + \lambda \sum_{(i,j)\in \text{MST}_{\mathscr{X}\setminus\{x_0\}}, i<j} D_{tri}[i,j,0](l_i, l_j, l_0) \tag{3}$$

The first term requires that the matched points should have similar SC. The second term regularizes the deformation of triangles. $\lambda$ ($\lambda \geq 0$) is a constant used to balance the weights of the two terms.

In the first term, $D_{sc}[i,0](l_i, l_0)$ denotes the distance between the oriented SC of $x_i$ and the oriented SC of $y_{l_i}$ measured by the $\chi^2$ test statistic, as presented in Eq. (1). The positive x-axis of SC for $x_i$ is directed at $x_0$, and the positive x-axis of SC for $y_{l_i}$ is directed at $y_{l_0}$. The computation of $D_{sc}[i,0](l_i, l_0), \forall i, l_i, l_0$, has time complexity $O(nm^2)$ and space complexity $O(nm^2)$. If the speedup measure proposed in subsection III-C is adopted, the time complexity is essentially $O(nmM)$.

In the second term, $D_{tri}[i,j,0](l_i, l_j, l_0)$ measures how far the affine transformation determined by correspondences $(i, j, 0) \to (l_i, l_j, l_0)$ is from a similarity transformation. We adopt the measure proposed in [30]. Since either the correspondences $(i, 0) \to (l_i, l_0)$ or the correspondences $(j, 0) \to (l_j, l_0)$ determine a similarity transformation, $D_{tri}[i,j,0](l_i, l_j, l_0)$ is defined as the $l_1$ norm distance between the parameters of the two transformations. Correspondences $(i, j) \to (l_i, l_j)$ also determine a similarity transformation. But we do not use them here because the length of the boundary edge $(x_i, x_j)$ is small and positional jitter of $x_i$ or $x_j$ may lead to significant change of similarity transformation, which will make the algorithm less robust.

### E. Algorithm

Since $G_{\mathscr{X}}$ is the result of connecting a single vertex $v_0$ to vertices in tree $\text{MST}_{\mathscr{X}\setminus\{x_0\}}$, $G_{\mathscr{X}}$ looks like a tree very much. Based on this fact, the DP algorithm on a tree [34] can be adapted to our optimization problem. In the following, we focus on the tree $\text{MST}_{\mathscr{X}\setminus\{x_0\}}$ whose vertices are $\{v_1, v_2, \dots, v_n\}$. Let $v_r$ be an arbitrary root vertex. The depth $d_i$ of $v_i$ is defined as the number of edges between $v_i$ and the root $v_r$ (the depth of $v_r$ is 0). The children $C_i$ of $v_i$ are the vertices adjacent to $v_i$ and having depth $d_i + 1$. Every vertex $v_i$ other than the root has a unique parent, which is the vertex adjacent to $v_i$ and having depth $d_i - 1$.

We define $n$ tables $V[i,0], i = 1, \dots, n$, each having $m \times m$ entries. The element $V[i,0](l_i, l_0)$ denotes the cost of the best placements of $v_0$, $v_i$ and the descendents of $v_i$ with the constraint that the placements of $v_0$ and $v_i$ are $l_0$ and $l_i$, respectively. Table $V[i,0]$ satisfies the following recursive equation:

$$V[i,0](l_i, l_0) = D_{sc}[i,0](l_i, l_0) + \sum_{j\in C_i} \min_{l_j} V[j,0](l_j, l_0) + \lambda D_{tri}[i,j,0](l_i, l_j, l_0) \tag{4}$$

For vertices with no children, we have

$$V[i,0](l_i, l_0) = D_{sc}[i,0](l_i, l_0) \tag{5}$$

For other vertices, the tables can be computed in terms of each other in descending depth order. Note that $V[r,0](l_r, l_0)$ is the cost of the best placement of the whole graph with the constraint that the placements of $v_r$ and $v_0$ are $l_r$ and $l_0$, respectively.

After all tables are computed, we can find the global minimum of the energy function by picking

$$(l_r^*, l_0^*) = \arg \min_{(l_r, l_0)} V[r,0](l_r, l_0) \tag{6}$$

and tracing back in order of ascending depth,

$$l_i^* = \arg \min_{l_i} V[i,0](l_i, l_0^*) + \lambda D_{tri}[i,j,0](l_i, l_j^*, l_0^*) \tag{7}$$

Here $v_j$ is the parent of $v_i$.

Since $G_{\mathscr{X}}$ is a 2-tree with the maximal cliques being triangles, and there are $n-1$ triangles, the above algorithm has time complexity $O(nm^3)$ and space complexity $O(nm^2)$. Similar to the practice in [12], we can reduce the running time of the algorithm based on the following observations. 1) Since the length of a boundary edge $(x_i, x_j)$ is small, given location $l_i$, the possible candidates for $l_j$ should be the indices of those points near $y_{l_i}$ (15 nearest points are chosen in our algorithm) because points far from it will introduce severe distortion in the model. 2) Given location $l_i$, the possible candidates for $l_0$ should be the indices of those points close to the circle centered at $y_{l_i}$ and with a radius equal to the length of the frame edge $(x_i, x_0)$ because points far from the circle will also introduce severe distortion in the model. With these two heuristics, the complexity of the algorithm is essentially $O(nm)$.

## IV. Shape representation based on star graph for point set matching

For the algorithm described in subsection III-E, if no speedup measure is taken, the time complexity will be $O(nm^3)$, which is relatively high. Therefore it will be of great interest if we could simplify the algorithm so that the time complexity can be reduced without sacrificing much the accuracy. Fortunately, this is possible. Due to the strong discriminative nature of SC, the frame edges with SCs acting as attributes in MST induced triangulation are already adequate to form a strong constraint on the shape of $\mathscr{X}$.

Here the shape representation for $\mathscr{X}$ is the same as that in subsection III-A except that no boundary edges are used. The graph $G'_{\mathscr{X}}$ for this shape representation is a *star* graph [11] with $v_0$ as the only internal vertex and all other vertices as leaves, as illustrated in Fig. 4. For $G'_{\mathscr{X}}$, the deformations of individual edges and the changes of angles between different edges will aggregate to generate the overall nonrigid deformation of $\mathscr{X}$, and our method is capable of handling nonrigid deformation to a certain extent.

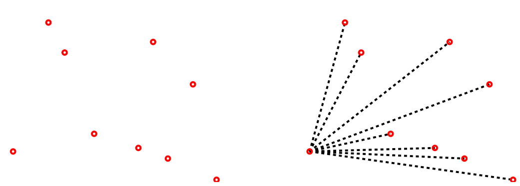

Fig. 4. An example of star graph. From left to right: a point set and its star graph.

The shape representation for $\mathscr{Y}$ is the same as that described in subsection III-C.

### A. Energy function

With the above discussion, the energy function takes the following form:

$$E(\phi) = \sum_{i=1}^{n} D_{sc}[i,0](l_i, l_0) + \mu \sum_{i=1}^{n} D_{frame}[i,0](l_i, l_0) \quad (8)$$

The first term is the same as that in Eq. (3). The second term requires that the lengths of edges in model graph should be preserved during matching. $\mu$ ($\mu \geq 0$) is a constant to balance the two terms.

In the second term, $D_{frame}[i,0](l_i, l_0)$ denotes the length difference between edge $(x_i, x_0)$ and the candidate edge $(y_{l_i}, y_{l_0})$. Based on the observation that shorter edges are less distorted than longer edges under a nonrigid deformation, the length differences of shorter edges should be penalized more than those of longer edges, and thus we use the $\chi^2$ test statistic instead of the Euclidean distance to measure the length difference:

$$D_{frame}[i,0](l_i, l_0) = \frac{|\|y_{l_i} - y_{l_0}\| - \|x_i - x_0\||^2}{\|y_{l_i} - y_{l_0}\| + \|x_i - x_0\|} \quad (9)$$

### B. Algorithm

Since $G'_{\mathscr{X}}$ is a tree, the DP algorithm on a tree [34] can be applied to our optimization problem. In addition, since $G'_{\mathscr{X}}$ is a star graph, it's convenient to take the internal vertex $v_0$ as the root of $G'_{\mathscr{X}}$. We define a table $V[0]$ with $m$ entries such that $V[0](l_0)$ denotes the best placements of all vertices with the constraint that the placement of $v_0$ is $l_0$. $V[0]$ satisfies the following equation:

$$V[0](l_0) = \sum_i \min_{l_i} D_{sc}[i,0](l_i, l_0) + \mu D_{frame}[i,0](l_i, l_0) \quad (10)$$

After $V[0]$ is computed, we can find the global minimum of the energy function by picking $l_0^* = \arg\min_{l_0} V[0](l_0)$ and tracing back as follows

$$l_i^* = \arg\min_{l_i} D_{sc}[i,0](l_i, l_0^*) + \mu D_{frame}[i,0](l_i, l_0^*) \quad (11)$$

Since $G'_{\mathscr{X}}$ is a tree with $n-1$ edges, the above algorithm has time complexity $O(nm^2)$ and space complexity $O(nm)$. Our experimental results indicate that the algorithm is very fast, and its running time is only a fraction of the time for computing the SC distances.

### C. Spatial mapping smoothing

For the algorithm described in subsection IV-B, since there is no constraint on angles between different edges, it may happen that points close to each other are mapped to points far away from each other. An example is illustrated in the left of Fig. 5. This problem can be remedied based on the following observation. Because $v_0$ is the conjunction of all edges in $G'_{\mathscr{X}}$, the matching of $v_0$ will affect the matching of all $n$ edges in $G'_{\mathscr{X}}$. In contrast, the matching of any vertex $v_i, i \neq 0$, will only affect the matching of the edge $(v_i, v_0)$. The algorithm in subsection IV-B is optimal. Among the locations $l_i, i = 0, \ldots, n$, outputted by the algorithm, $l_0$ is $n-1$ times more reliable than any other location. Inspired by this observation, we can make the spatial mapping smoother via the following steps. 1) Run the algorithm described in subsection IV-B to obtain location $l_0$ and discard other locations $l_i, i = 1, \ldots, n$. 2) Find locations $l_i, i = 1, \ldots, n$, by algorithm described in subsection III-E with $l_0$ predetermined. Because the algorithm in subsection III-E preserves lengths of edges in $\text{MST}_{\mathscr{X}\setminus\{x_0\}}$, the resulting spatial mapping will be smooth.

With $l_0$ predetermined, the problem of finding the best embedding of $G_{\mathscr{X}}$ in $\mathscr{Y}$ reduces to the problem of finding the best embedding of $\text{MST}_{\mathscr{X}\setminus\{x_0\}}$ in $\mathscr{Y}$. Since $\text{MST}_{\mathscr{X}\setminus\{x_0\}}$ is a tree, the algorithm in subsection III-E with $l_0$ predetermined has time complexity $O(nm^2)$ and space complexity $O(nm)$, the same as those of the algorithm in subsection IV-B. The difference of matching results before and after spatial mapping smoothing is illustrated in Fig. 5 by using an example.

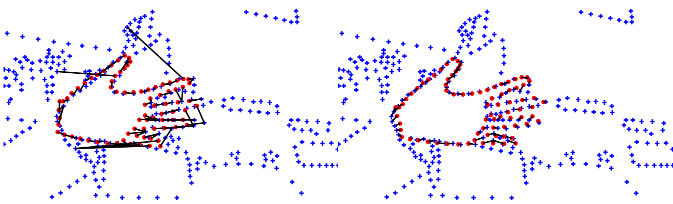

Fig. 5. An example of point set matching before (left) and after spatial mapping smoothing (right). The data points are shown as blue $+$. The affinely transformed model points are shown as red $*$. Point correspondences are indicated by black line segments.

## V. Experimental results

We compare our methods based on MST induced triangulation (denoted by MSTT) and on star graph (denoted by SG) with the following representative point set matching methods: the local neighborhood structure preserving (LNSP) method [9], the Viterbi algorithm (VA) based method [27], the linear programming (LP) based method [23] and the constrained Delaunay triangulation (CDT) based method [13]. Since CDT can only be used for template matching in images, it is only tested in subsection V-C where images are involved. We also compare MSTT with our previously proposed fan-shaped triangulation (FST) based method [12] in subsection V-A. To make a fair comparison, the deformation regularization terms in FST are changed to the second term in Eq. (3). LP is a general matching algorithm and different feature descriptors can be used. We use SC as the feature descriptor for LP. VA and LP are not rotation invariant, and we render them rotation invariant by evaluating them on 8 evenly quantized angles and choosing the result with the minimum cost.

We implement all the competing methods [1] under Matlab (version 7.6) environment on a PC with 2.4GHz CPU and 3G memory. We use affine transformation to model a nonrigid spatial mapping. Correspondence recovered by one method is used to solve for affine transformation. First we use synthetic data to evaluate various aspects of the methods. Then we compare the methods using data acquired from real images.

### A. *Experiments against deformation, noise and outliers*

Synthetic data can be used to test specific aspects of an algorithm. To evaluate the improvement of MSTT over FST comprehensively, we use two categories of shapes to generate model point sets: 1) the 99 silhouettes provided by Kimia [35], which resemble simple polygons, and 2) 100 Chinese characters extracted from the HIT-MW database [36], which are more complex. Fig. 6 shows the two categories of shapes.

Fig. 6. Shapes used to generate model point sets in the experiments against deformation, noise and outliers. Left column: Kimia's 99 silhouette dataset. Right column: Chinese characters extracted from the HIT-MW database.

A procedure similar to the generation of Chui-Rangarajan data set [1] is used to generate a series of data point sets for each model shape. 1) The model shape was randomly rotated and then nonrigidly deformed to generate the data point set. Gaussian radial basis functions (RBF) were used to generate nonrigid deformations with coefficients sampled from a Gaussian distribution of zero mean and standard deviation ranging from 0.01 to 0.05. The aim is to test one method's robustness against nonrigid deformation. 2) Random positional noise (which is generated from Gaussian noise with standard deviation ranging from 0.01 to 0.05) was added to a randomly rotated and then moderately nonrigidly deformed model shape. The aim is to test one method's robustness against noise. 3) Random outliers (outlier ratio ranging from 0.5 to 2.5) were added to a randomly rotated and then moderately nonrigidly deformed model shape. The aim is to test one method's robustness against outliers. Fig. 7 shows examples of model point sets and their corresponding data point sets in the 3 series of tests, respectively.

Fig. 7. Examples of model point sets (left column) and data point sets in the deformation, noise and outlier tests, respectively (right 3 columns).

The average errors by various methods are shown in Fig 8. Here error is defined as the mean of the Euclidean distance between the affinely transformed model points and their corresponding ground truth data points. From the figure, it can be seen that MSTT, SG and FST perform the best for the deformation and outlier tests, and they perform in average compared with other methods for the noise test. The outlier test witnesses the most significant difference between

[1]The source codes of the proposed mehtods can be downloaded at `http://www4.comp.polyu.edu.hk/∼cslzhang/code.htm`.

these 3 methods and other methods, where there is a large margin between their results. This experiment demonstrates the robustness of MSTT, SG and FST to various types of disturbances, particularly outliers. Among these 3 methods, it can be seen than MSTT performs the best, FST the second, and SG the third.

The average running time of various methods is listed in Table I. It can be seen that the running time of MSTT, FST and SG is low when the number of points is small. When the number of points becomes large (i.e., in the case of outliers), SG is still among the fastest methods. In comparison, MSTT's running time is in average compared with other methods and FST is the slowest among all the methods. This demonstrates SG's high computational efficiency.

TABLE I
AVERAGE RUNNING TIME (SECONDS)

| | Deformation | Noise | Outliers |
|---|---|---|---|
| MSTT | 6.2470 | 6.4055 | 24.1584 |
| SG | **3.9627** | **3.9843** | 12.0901 |
| FST | 6.3847 | 6.1566 | 44.9609 |
| VA | 4.9842 | 5.0315 | **10.9152** |
| LP | 16.7982 | 15.8847 | 23.4747 |
| LNSP | 4.2650 | 5.1858 | 18.9177 |

The parametric robustness of MSTT and SG are illustrated in Fig. 9. Here we only use the outlier test with outlier ratio being 2.5 for testing. From the figure, it can be seen that SG's error depends mainly on the value of $\lambda$, while it is insensitive to the value of $\mu$. SG achieves the lowest error when $(\lambda, \mu) = (1, 3)$ for Kimia's 99 silhouette test or when $(\lambda, \mu) = (1, 2)$ for the 100 Chinese character test. MSTT achieves the lowest error when $\lambda = 6$ for Kimia's 99 silhouette test or when $\lambda = 2$ for the 100 Chinese character test.

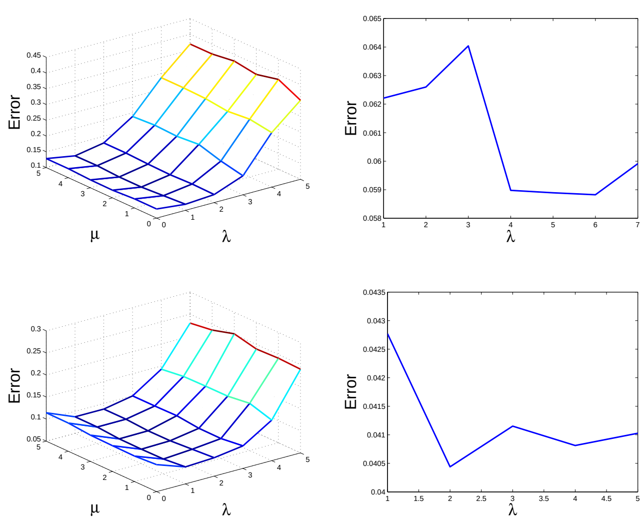


Fig. 9. Average errors of SG (left column) and MSTT (right column) with respect to different parameter values in the outlier test with outlier ratio being 2.5. Top row: Kimia's 99 silhouette dataset. Bottom row: the 100 Chinese character dataset.

## B. Experiment against clutter

We then test the robustness of various methods against clutter. Two shapes similar to the model (the first one is called the ground truth target shape because it is similar to the model, and the second one is called the disturbance shape which is generated from the model by a mirror reflection) are mixed together (with the disturbance shape rotated in $\left[0, \frac{1}{100}2\pi, \ldots, \frac{99}{100}2\pi\right]$ ) to generate the data point set. Some examples of the data point set generation are illustrated in Fig. 10. The aim of this set up is to animate complex clutter. Error is defined as the Hausdorff distance between the affinely transformed model by a method and the ground truth target shape. The average errors of various methods are shown in Fig. 11. It can be seen that MSTT performs the best. SG performs slightly worse than MSTT but better than the rest methods. LNSP performs the worst among all the methods. Examples of matching results by various methods are shown in Fig. 12. It can be found that MSTT and SG are more robust to clutter than other methods.

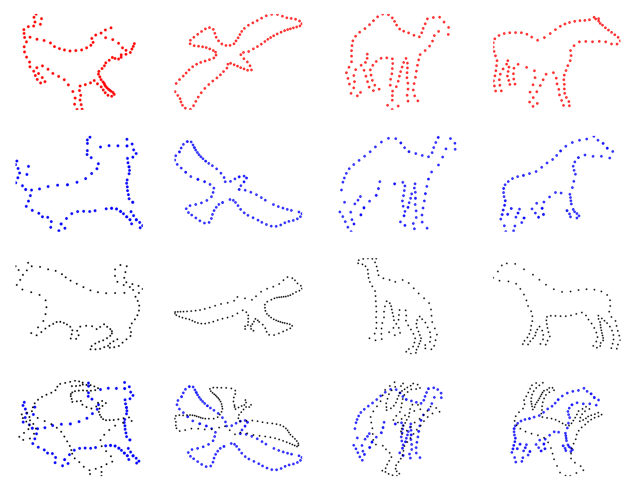

Fig. 10. Set up of the experiment against clutter. From top row to bottom row: 1) model shapes; 2) ground truth target shapes; 3) disturbance shapes; 4) the mixture of ground truth target shapes and rotated disturbance shapes.

## C. Experiment on the ETHZ data set

We finally test the performance of various methods on the ETHZ data set [37], which consists of 255 images of 5 categories: apple logo, swan, giraffe, bottle and mug. The models for the 5 categories of images are shown in Fig. 13. The objects in each image are often substantially different from the corresponding model. Also, there are clutters in the images. These facts make the data set a good test platform for deformable matching with clutters. Each image has a ground truth target shape (some have multiple ground truth target shapes and we only used the largest one). Error is defined as the Hausdorff distance between the affinely transformed model by a method and the ground truth target shape. Because SC is not scale invariant, we scaled both the model shape and the ground truth target shape (with the image containing it scaled accordingly) to be unit sized. To generate model and data point sets, we sampled 70 points from a model and 300

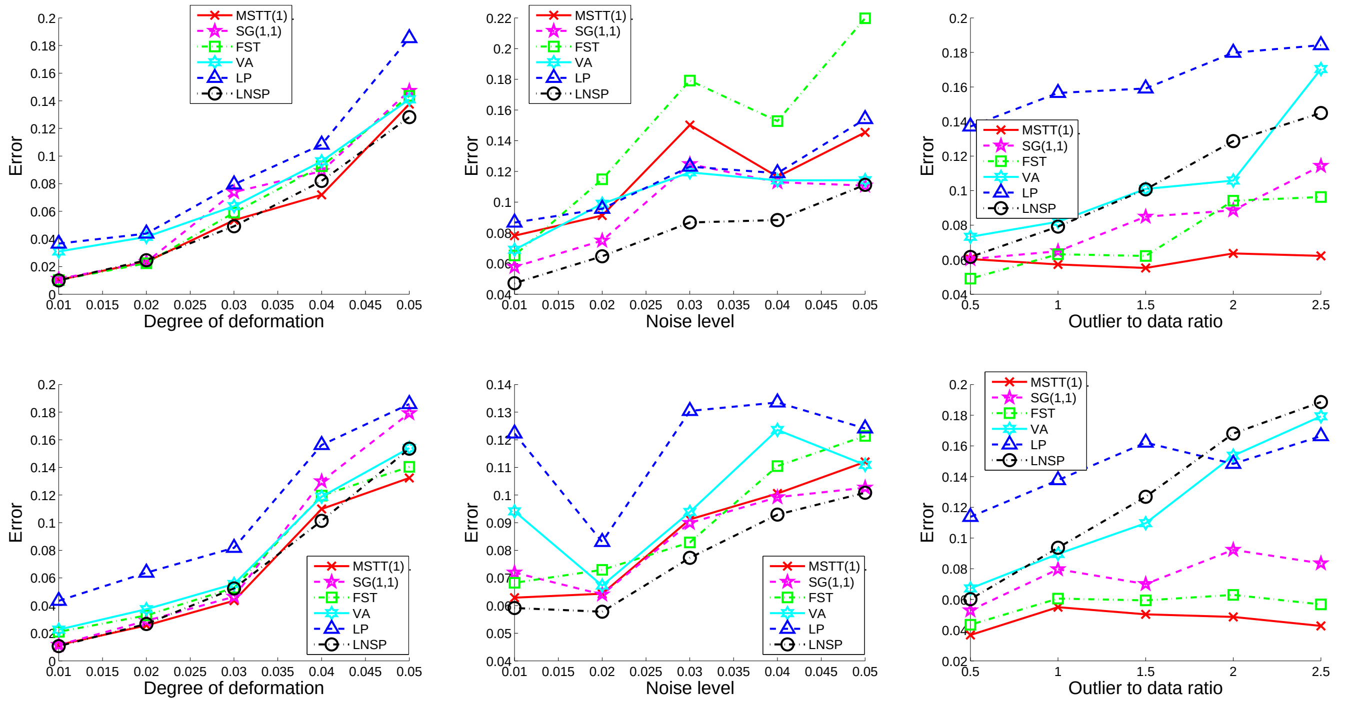


Fig. 8. Average errors of MSTT with $\lambda = 1$, SG with $(\lambda, \mu) = (1, 1)$, FST with $\lambda = 1$, VA, LP and LNSP in the experiment against deformation, noise and outlier. Top row: Kimia's 99 silhouette dataset. Bottom row: the 100 Chinese character dataset.

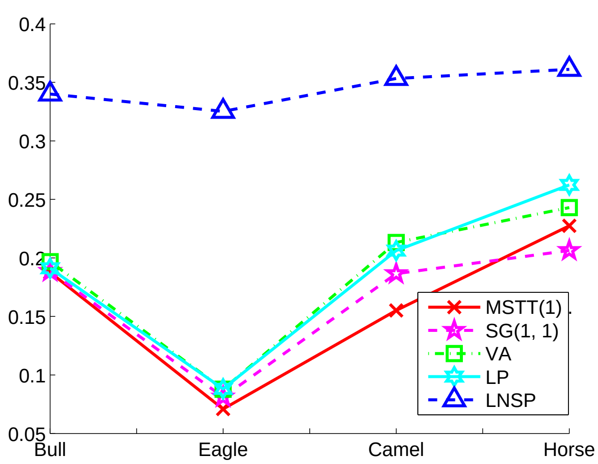


Fig. 11. Average errors of MSTT with $\lambda = 1$, SG with $(\lambda, \mu) = (1, 1)$, VA, LP and LNSP in the experiment against clutter.

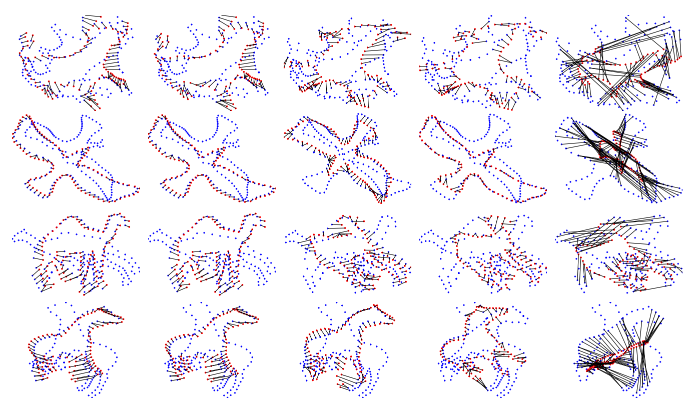

Fig. 12. Examples of matching results by MSTT (left column), SG (column 2), VA (column 3), LP (column 4) and LNSP (right column) in the experiment against clutter. The affinely transformed model points are shown as red $*$. Point correspondences are indicated by black line segments.

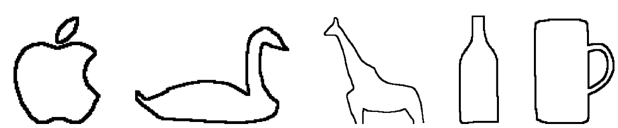

Fig. 13. The model shapes used for matching in the ETHZ data set.

points from the edges of an image [38]. The average errors of various methods are shown in Fig. 14. It can be seen that MSTT performs the best. SG performs similarly to VA and LP. LNSP performs the worst among all the methods, and CDT's performance is in between LNSP and other methods. Examples of matching results by various methods are shown in Fig. 15 to Fig. 19. One can see that MSTT matches a model more tightly to the corresponding target shape than other methods, as is evident for the swan and giraffe tests.

## VI. Conclusion

To address the problem of rotation invariant nonrigid point set matching, we proposed two methods for shape representation. The shape context (SC) feature descriptor was used and we constructed graphs on point sets where edges are used to determine the orientations of SCs. This enables the proposed methods rotation invariant. The structures of our

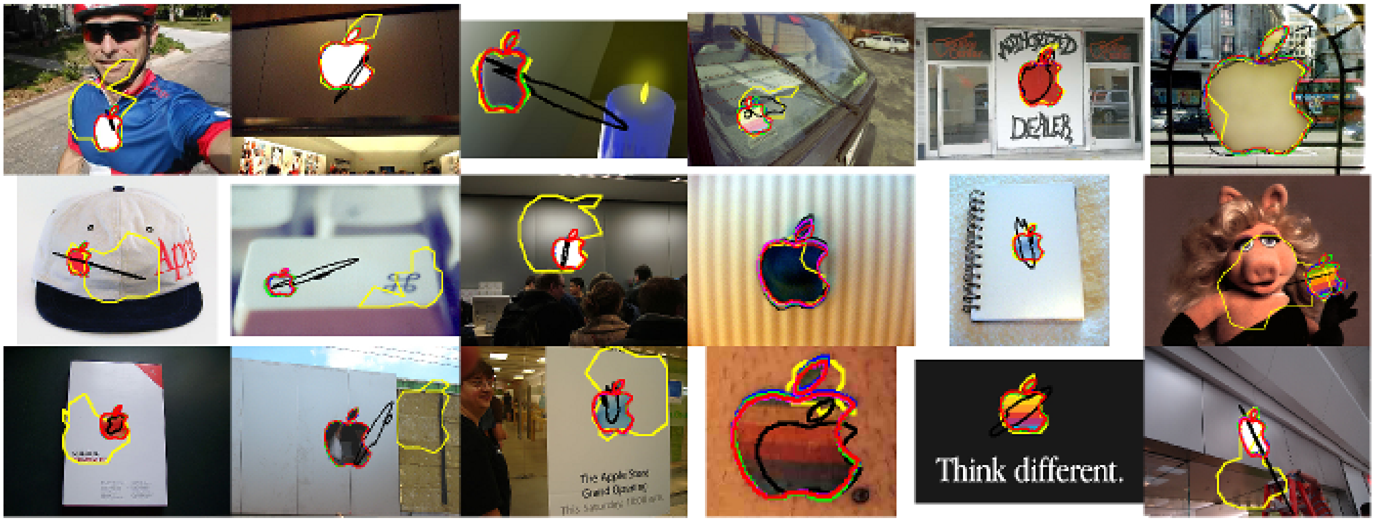


Fig. 15. Examples of matching results by MSTT (red), SG (magenta), VA (green), LP (blue), LNSP (black) and CDT (yellow) on the apple logo images from the ETHZ dataset. The matching results are indicated by affinely transformed model shapes by various methods.

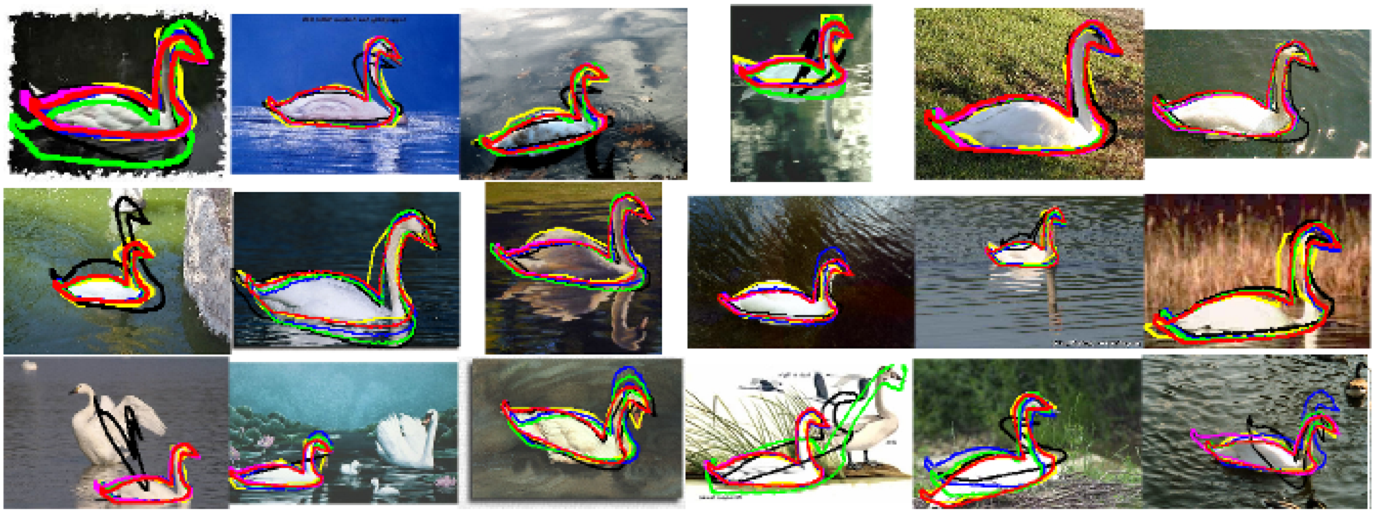

Fig. 16. Examples of matching results by MSTT (red), SG (magenta), VA (green), LP (blue), LNSP (black) and CDT (yellow) on the swan images from the ETHZ dataset. The matching results are indicated by affinely transformed model shapes by various methods.

shape representations facilitate the use of DP for optimization. The strong discriminative nature of SC, the calculated robust orientations of SCs, and the global optimality of DP make our methods robust to various types of disturbances, particularly clutters.

The proposed methods were tested on both synthetic and real data in comparison with several representative methods. The results show that our methods, especially MSTT, clearly outperform other methods in terms of robustness against clutter. The proposed methods are very useful for tasks involving detection and matching of shapes in cluttered scenes where the initial poses of the shapes may not be known.

## References


[1] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understanding*, vol. 89, no. 2-3, pp. 114–141, 2003.

[2] R. C. Veltkamp and M. Hagedoorn, "State of the art in shape matching," in *Principles of visual information retrieval*. London, UK: Springer-Verlag, 2001, pp. 87–119.

[3] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, 2002.

[4] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 2004.

[5] W. Cheung and G. Hamarneh, "N-sift: N-dimensional scale invariant feature transform," *IEEE Trans. Image Processing*.

[6] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, 2005.

[7] Z. Chen and S.-K. Sun, "A zernike moment phase-based descriptor for local image representation and matching," *IEEE Trans. Image Processing*, vol. 19, no. 1, pp. 205–219, 2010.

[8] H. Jiang and S. X. Yu, "Linear solution to scale and rotation invariant object matching," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2009, pp. 2474–2481.

[9] Y. Zheng and D. Doermann, "Robust point matching for nonrigid shapes by preserving local neighborhood structures," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 643–649, 2006.

[10] D. B. West, "Introduction to graph theory," 2001, pearson Education, Inc.

[11] C. Godsil and G. Royle, "Algebraic graph theory," 2001, springer-Verlag new York, Inc.

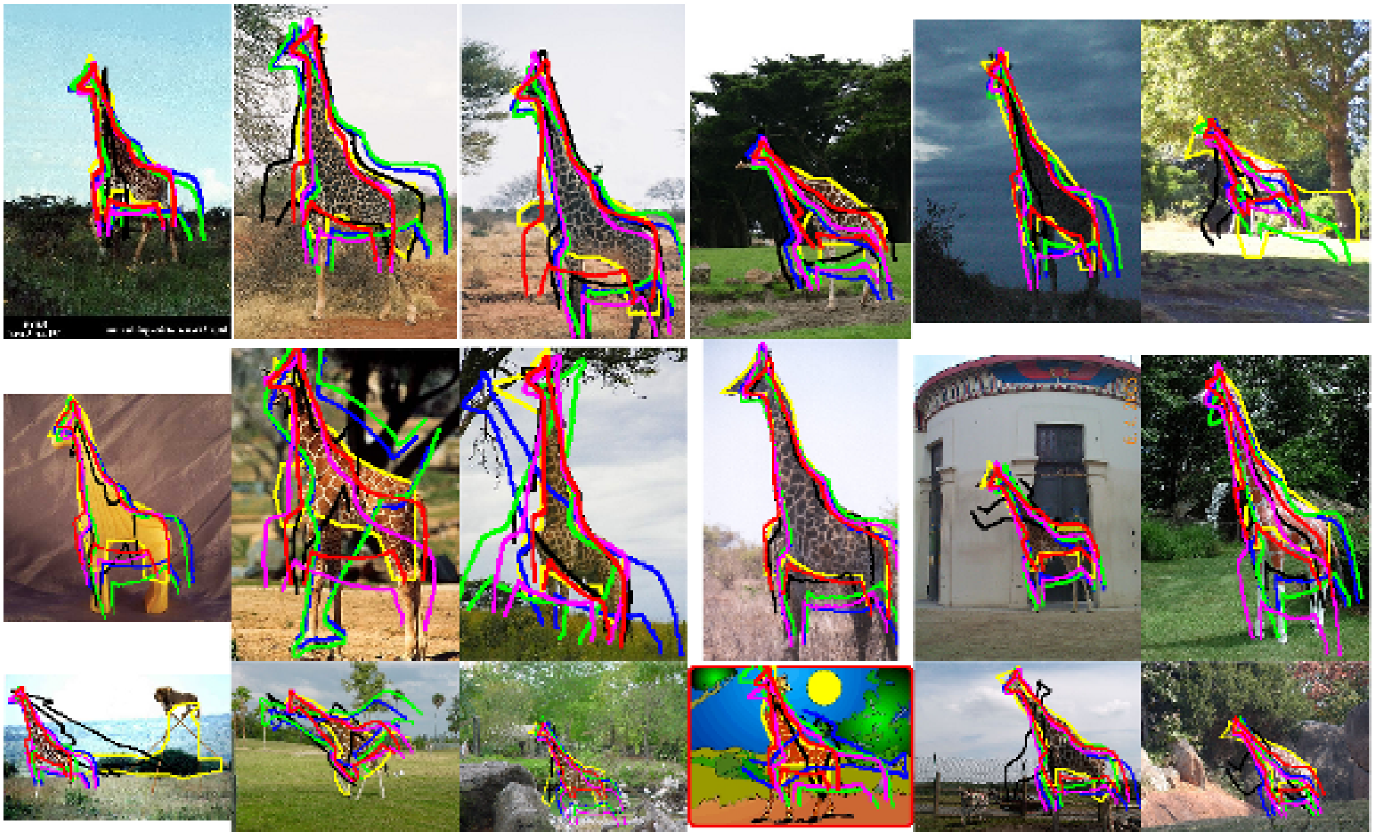

Fig. 17. Examples of matching results by MSTT (red), SG (magenta), VA (green), LP (blue), LNSP (black) and CDT (yellow) on the giraffe images from the ETHZ dataset. The matching results are indicated by affinely transformed model shapes by various methods.

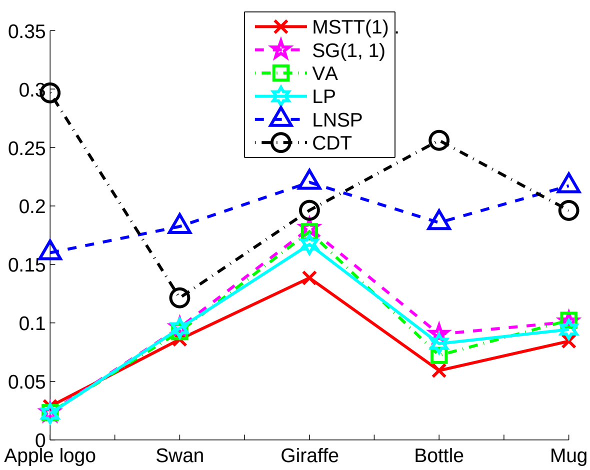


Fig. 14. Average errors of MSTT with $\lambda = 1$, SG with $(\lambda, \mu) = (1, 1)$, VA, LP, LNSP and CDT on the ETHZ data set.

[12] W. Lian and L. Zhang, "Rotation invariant non-rigid shape matching in cluttered scenes," in *European Conference on Computer Vision*, 2010.
[13] P. F. Felzenszwalb, "Representation and detection of deformable shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 208–220, 2005.
[14] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, "Computational geometry," 1997, springer-Verlag.
[15] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 377–388, 1996.
[16] M. Leordeanu and M. Hebert, "A spectral technique for correspondence problems using pairwise constraints," in *IEEE International Conference on Computer Vision*, 2005.
[17] T. Cour, P. Srinivasan, and J. Shi, "Balanced graph matching," in *Advances in Neural Information Processing Systems (NIPS)*, 2006.
[18] A. Egozi, Y. Keller, and H. Guterman, "Improving shape retrieval by spectral matching and meta similarity," *IEEE Trans. Image Processing*, vol. 19, no. 5, pp. 1319–1327, 2010.
[19] C. Schellewald and C. Schnorr, "Probabilistic subgraph matching based on convex relaxation," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2005.
[20] L. Torresani, V. Kolmogorov, and C. Rother, "Feature correspondence via graph matching: Models and global optimization," in *European Conference on Computer Vision*, 2008, pp. 596–609.
[21] M. Torki and A. Elgammal, "One-shot multi-set non-rigid feature-spatial matching," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2010.
[22] H. Wang, S. Yan, J. Liu, X. Tang, and T. S. Huang, "Correspondence propagation with weak priors," *IEEE Trans. Image Processing*, vol. 18, no. 1, pp. 140–150, 2009.
[23] H. Jiang, M. S. Drew, and Z.-N. Li, "Matching by linear programming and successive convexification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 959–975, 2007.
[24] H. Li, E. Kim, X. Huang, and L. He, "Object matching with a locally affine-invariant constraint," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2010.
[25] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.
[26] C. Scott and R. D. Nowak, "Robust contour matching via the order-preserving assignment problem," *IEEE Trans. Image Processing*, vol. 15, no. 7, pp. 1831–1838, 2006.
[27] A. Thayananthan, B. Stenger, P. H. S. Torr, and R. Cipolla, "Shape context and chamfer matching in cluttered scenes," in *IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, 2003, pp. 127–133.
[28] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE Trans. Information Theory*, vol. 46, no. 2, pp. 325–343, 2000.

Fig. 18. Examples of matching results by MSTT (red), SG (magenta), VA (green), LP (blue), LNSP (black) and CDT (yellow) on the bottle images from the ETHZ dataset. The matching results are indicated by affinely transformed model shapes by various methods.

Fig. 19. Examples of matching results by MSTT (red), SG (magenta), VA (green), LP (blue), LNSP (black) and CDT (yellow) on the mug images from the ETHZ dataset. The matching results are indicated by affinely transformed model shapes by various methods.

[29] T. S. Caetano, T. Caelli, D. Schuurmans, and D. A. Barone, "Graphical models and point pattern matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1646–1663, 2006.
[30] T. S. Caetano and T. Caelli, "A unified formulation of invariant point pattern matching," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 3, 2006, pp. 121–124.
[31] J. J. McAuley and T. S. Caetano, "Exploiting within-clique factorizations in junction-tree algorithms," *Journal of Machine Learning Research*, vol. 9, pp. 525–532, 2010.
[32] P. F. Felzenszwalb and J. J. McAuley, "Fast inference with min-sum matrix product," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 99, no. PrePrints, 2011.
[33] Http://en.wikipedia.org/wiki/Traveling_salesman_problem.
[34] P. F. Felzenszwalb and R. Zabih, "Dynamic programming and graph algorithms in computer vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, pp. 721–740, 2011.
[35] T. B. Sebastian, P. N. Klein, and B. B. Kimia, "Recognition of shapes by editing their shock graphs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 550–571, 2004.
[36] T. Su, T. Zhang, and D. Guan, "Corpus-based hit-mw database for offline recognition of general-purpose chinese handwritten text," *International Journal of Document Analysis and Recognition*, vol. 10, no. 1, pp. 27–38, 2007.
[37] V. Ferrari, T. Tuytelaars, and L. V. Gool, "Object detection by contour segment networks," in *ECCV*, 2006.
[38] D. Martin, C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 530–549, 2004.


**Wei Lian** received the B.S. degree in automation from Taiyuan University of Technology, Taiyuan, P.R. China, in 2000. He received the M.S. and Ph.D degrees in Automatic Control Theory and Engineering from Northwestern Polytechnical University, Xi'an, P.R. China, respectively in 2003 and 2007. Since 2010, he has been with the Department of Computer Science, Changzhi University, Changzhi, China. His research interests include Optimization Theory and Computer Vision.

**Lei Zhang (M'04)** received the B.S. degree in 1995 from Shenyang Institute of Aeronautical Engineering, Shenyang, P.R. China, the M.S. and Ph.D degrees in Automatic Control Theory and Engineering from Northwestern Polytechnical University, Xi'an, P.R. China, respectively in 1998 and 2001. From 2001 to 2002, he was a research associate in the Dept. of Computing, The Hong Kong Polytechnic University. From Jan. 2003 to Jan. 2006 he worked as a Postdoctoral Fellow in the Dept. of Electrical and Computer Engineering, McMaster University, Canada. In 2006, he joined the Dept. of Computing, The Hong Kong Polytechnic University, as an Assistant Professor. Since Sept. 2010, he has been an Associate Professor in the same department. His research interests include Image and Video Processing, Biometrics, Computer Vision, Pattern Recognition, Multisensor Data Fusion and Optimal Estimation Theory, etc. Dr. Zhang is an associate editor of IEEE Trans. on SMC-C, IEEE Trans. on CSVT and Image and Vision Computing Journal. Dr. Zhang was awarded the Faculty Merit Award in Research and Scholarly Activities 2010, and the Best Paper Award of SPIE VCIP2010. More information can be found in his homepage http://www4.comp.polyu.edu.hk/∼cslzhang/.

**David Zhang (F'09)** graduated in computer science from Peking University in 1974 and received his M.Sc. and Ph.D. degrees in Computer Science and Engineering from the Harbin Institute of Technology (HIT), Harbin, P.R. China, in 1983 and 1985, respectively. He received the second Ph.D. degree in Electrical and Computer Engineering at the University of Waterloo, Waterloo, Canada, in 1994. From 1986 to 1988, he was a Postdoctoral Fellow at Tsinghua University, Beijing, China, and became an Associate Professor at Academia Sinica, Beijing, China. Currently, he is a Professor with the Hong Kong Polytechnic University, Hong Kong. He is Founder and Director of Biometrics Research Centers supported by the Government of the Hong Kong SAR (UGC/CRC). He is also Founder and Editor-in-Chief of the International Journal of Image and Graphics (IJIG), Book Editor, The Kluwer International Series on Biometrics, and an Associate Editor of several international journals. His research interests include automated biometrics-based authentication, pattern recognition, biometric technology and systems. As a principal investigator, he has finished many biometrics projects since 1980. So far, he has published over 200 papers and 10 books.