Pergamon

Pattern Recognition, Vol. 30, No. 2, pp. 311–320, 1997
Copyright © 1997 Pattern Recognition Society. Published by Elsevier Science Ltd
Printed in Great Britain. All rights reserved
0031–3203/97 $17.00+.00

PII:S-0031-3203(96)00076-3

FAST ALGORITHM FOR POINT PATTERN MATCHING: INVARIANT TO TRANSLATIONS, ROTATIONS AND SCALE CHANGES

SHIH-HSU CHANG,[†,*] FANG-HSUAN CHENG,[‡] WEN-HSING HSU[†,§] and GUO-ZUA WU[†]
[†]Department of Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan 300, R.O.C.
[‡]Department of Computer Science, Chung-Hua Polytechnic Institute, Hsinchu, Taiwan 300, R.O.C.
[§]Institute of Information Science, Academia Sinica, Taiwan, R.O.C.

(*Received* 12 *December* 1995; *received for publication* 1 *April* 1996)

Abstract—Based on 2-D cluster approach, a fast algorithm for point pattern matching is proposed to effectively solve the problems of optimal matches between two point pattern under geometrical transformation and correctly identify the missing or spurious points of patterns. Theorems and algorithms are developed to determine the matching pairs support of each point pair and its transformation parameters (scaling s and rotation θ) on a two-parameter space (s,θ). Experiments are conducted both on real and synthetic data. The experimental results show that the proposed matching algorithm can handle translation, rotation, and scaling differences under noisy or distorted condition. The computational time is just about 0.5 s for 50 to 50 point matching on Sun-4 workstation. Copyright © 1997 Pattern Recognition Society. Published by Elsevier Science Ltd.

Point pattern matching   Affine transformation   Inexact matching   Registration
Maximum matching pairs support   Hough transform

1. INTRODUCTION

Point pattern matching plays an important role in pattern recognition and computer vision.[1–12] It adopts to find pairs of point features in two images taken from a scene and that makes each pair corresponds to the same scene point. Feature points could be curve intersections, ending points of curves, corners, high curvature points, zero-crossings, etc. This paper focuses on the problem of point pattern matching by using the geometric information under the assumption that the feature points are unlabeled and the correspondence between two point patterns is unknown.

There are several difficulties associated with point pattern matching:

1. Two images of the same object may be obtained from different view points, from different sensors, or from the same sensor at different times. Therefore, two point patterns may have translation, rotation, and scaling differences. Most of the time, however, these geometrical quantities are not known. If the corresponding points are known, the geometrical transformation parameters (translation, rotation, and scaling) can be found by using least-squares technique to estimate the least squared error between these corresponding points.[13,14]
2. Due to noise or the feature extracted process, the relative position of points may have perturbations, called local distortion. A registration that exactly superimposes one point pattern on another point pattern may not exist, so the approximate point pattern matching is needed. The method is to find a registration transformation that puts a point in one point pattern within the noisy region of its corresponding point in another point pattern.[13]
3. An image may consists of sub-parts of the model object; thus, these can be extra points in one point pattern. An exact one-to-one correspondence between two point patterns does not often exist. Therefore, inexact matching is needed. In other words, we give two point patterns $P=\{p_1, p_2,\dots,p_m\}$ and $Q=\{q_1,q_2,\dots,q_n\}$, it is to find the maximum matching pairs k between P and Q, where k is unknown and $k\le\min(m,n)$.
4. The process of finding matching pairs is time consuming. A fast and effective algorithm is necessary.

Therefore, for a good point pattern matching approach, it is able to perform the geometrical invariant quantities (translation, rotation, and scaling) efficiently and match well under translation, rotation, scaling, local distortion, and extra/missing points conditions. The goal of this paper is to develop such an algorithm.

Various algorithms have been proposed for point matching by using different methodologies. Lavine *et al.*[15] use minimal spanning trees and nearest

[*] Author to whom the correspondence should be addressed. Tel.: 886-35-710300; fax: 886-35-715971; e-mail: whhsu@-mercury.ee.nthu.edu.tw.

neighbour distances to form a measure of similarity between point patterns under the translation and rotation. Ogawa(16) uses the Delaunay triangulation and maximal clique to obtain the largest set of mutually consistent pairs between labeled point patterns under affine transformation. Hong and Tan(17) propose a method for transforming a set of n points to a canonical form in $O(n)$ time and an average $O(n)$ time algorithm to find a matching between canonical forms of two point sets under affine transformation, where n is the number of points in the two images. If the number of matching pairs is not n, as in our case, the complexity of this algorithm becomes exponential.

An asymmetric neural network for point matching proposed by Vinod and Ghose(18) can handle translation, rotation, distortion, and noise but not scaling. When the network is simulated on a uniprocessor machine, it requires $O(n^3)$ neuron updations. For 50-to-50 point matching, the average time required by sequential simulations of the network in an IBM PC-AT is 8251 s and the number of average iterations is 290.

Some experiments on point pattern matching by using the relaxation approach are described in references (12,19–24). Ranade and Rosenfeld algorithm(19) handles translational difference only. Its complexity is $O(n^4)$. Wang *et al.*(12) and Ton and Jain's(20) algorithms can handle translational and rotational differences. The complexity of Ton's algorithm is $O(n^3)$. Li,(21) Jeng *et al.*'s(22) and Ogawa *et al.*'s(24) algorithms can handle translational, rotational and scaling differences. However, their complexities are at least $O(n^6)$ and the individual steps involved in the relaxation process are computationally intensive.

Skea *et al.*(25) deal with the dual problems of finding matches between two point sets. They use an accumulator to count possible matches by comparing all triangles formed by three points between two point sets. The accumulator is scanned to identify corresponding points. The algorithm can find matches between two similar point sets, but it may find incorrect matches.

Stockman *et al.*(26) use the clustering approach to estimate the transformation parameters (scaling, rotation, x translation, and y translation) and then determine the corresponding points between two point patterns. Correct matches tend to make a cluster. The parameter values correspond to the center of densest cluster. Goshtasby and Stockman(27) use the convex hull property to choose the subsets of points for matching in order to reduce the computational time of Stockman's algorithm. When two convex hull boundaries in two point patterns have some common points but have no common edge or no feature point in boundaries, they will result in mismatches.

In the cluster approach mentioned above, the parameter space is a four-dimension space (scaling s, rotation θ, x translation t_x, and y translation t_y. A 4-D clustering routine requires large memories and time to process. If noise and distortions are present, they could be in the form of larger variances of s, θ, t_x, and t_y. The larger variances produce a relatively broad peak. Many larger points make it difficult to find the exact position of the peak (the registration parameters). The convex hull property is not suitable to reduce the computational time when there are many extra or missing point patterns.

In our approach, theorems and algorithms are developed to determine the matching pairs support of each point pair and its transformation parameters (scaling s and rotation θ) on a two-parameter space (s,θ). Since the parameter space just considers s and θ, the effect of the variances of t_x and t_y can be removed. It is easy and fast to decide the exact transformation parameters. For matching two point patterns P and Q under translation, rotation, scaling, local distortion, and extra/missing points conditions, the algorithm is stated as follows:

1. Detect the point pairs that own the maximum matching pairs support and the registration parameters (s,θ) which can match the most points between P and Q.
2. Determine the matching pairs under the registration parameters between P and Q.
3. Determine the optimal transformation parameters between matching pairs and minimize the sum of squared coordinated differences.

The detail algorithms are presented in Sections 2 and 3. Section 4 describes the experimental results and the performance of the algorithm. Finally, the conclusion is discussed in Section 5.

2. DETERMINATION OF THE MAXIMUM MATCHING PAIRS SUPPORT

In this section, we develop the theorems and algorithms which can simultaneously estimate the transformation parameters and the maximum correct number of matched pairs. After the transformation parameters are estimated, the matching pairs which satisfy the consistent condition are selected as the corresponding pairs.

2.1. *Problem formulation*

Two points patterns P and Q consists of m feature points extracted from the first image, where $P = \{p_1, p_2, \ldots, p_m\}$, and Q consists of n feature points extracted from the second image, where $Q = \{q_1, q_2, \ldots, q_n\}$. Each point is described with the coordinates $(x, y)^{\mathrm{T}}$, that is $P = \{(x_{p_i}, y_{p_i})^{\mathrm{T}} \mid i = 1, \ldots, m\}$ and $Q = \{(x_{q_a}, y_{q_a})^{\mathrm{T}} \mid a = 1, \ldots, n\}$. A matching is to find a correspondence between a point p_i and P and a certain point q_a in Q; that makes this corresponding pair consistent under a registration $G_{(t_x,t_y,s,\theta)}$. In this paper the registration $G_{(t_x,t_y,s,\theta)}$ is an affine transformation composed of four parameters: s,θ,t_x, and t_y, where s is a scaling factor, θ the rotation angle, and t_x and t_y the translations along the x and y directions, respectively. Let $r = (t_x, t_y, s, \theta)$. The registration maps straight lines into straight lines and maps a triangle into a similar triangle. It is a global transformation since the overall geometric relationships between points do not change. It maps a point $p = (x_p, y_p)^{\mathrm{T}}$ to a point $q = (x_q, y_q)^{\mathrm{T}}$ as follows:

$$q = G_r(p) \Rightarrow \begin{pmatrix} x_q \\ y_q \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} s\cos\theta & -s\sin\theta \\ s\sin\theta & s\cos\theta \end{pmatrix} \begin{pmatrix} x_p \\ y_p \end{pmatrix}. \tag{1}$$

Because the relative position of points may have perturbations, $G_r(p)$ may not exactly equal to q. Each corresponding pair $p_i \leftrightarrow q_a$ is said to be consistent if the coordinates of $G_r(p_i)$ differ with q_a within a given bound d_1, i.e. $\|G_r(p_i) - q_a\| \leq d_1$, where d_1 is the allowable distortion bound.

The objective of this paper is to find a registration so that the correct number of matched pairs between P and Q is maximum. The registration transformation puts each matched point in P within the distortion region (d_1) of its corresponding point in Q.

2.2. Theorems

There are two equations in equation (1), with four unknown variables t_x, t_y, s and θ, so there are infinite solutions satisfying equation (1). Thus, the registration is not unique. Let us give two pairs (p_i, p_j) in P and (q_a, q_b) in Q. If $p_i \neq p_j$ and $q_a \neq q_b$, then an unique registration $G_{(t_x,t_y,s,\theta)}$ exits; that makes $q_a = G(p_i)$ and $q_b = G(p_j)$. It is stated in the following theorem.

Theorem 1 (Unique registration). Let (p_i,p_j) and (q_a, q_b) be two pairs in P and Q. If $p_i \neq p_j$ and $q_a \neq q_b$, then a unique registration G_r exists; that makes $q_a = G_r(p_i)$, $q_b = G_r(p_j)$, where $r = (t_x, t_y, s, \theta)$, $s = |\overrightarrow{q_aq_b}| / |\overrightarrow{p_ip_j}|$, $\theta = \theta_{\overrightarrow{q_aq_b}} - \theta_{\overrightarrow{p_ip_j}}$, $t_x = x_{q_a} - x_{p_i}(s\cos\theta) + y_{p_i}(s\sin\theta)$ and $t_y = y_{q_a} - x_{p_i}(s\sin\theta) + y_{p_i}(s\cos\theta)$.

Proof. Assume that $q_a = G_r(p_i)$ and $q_b = G_r(p_j)$. Thus

$$\begin{pmatrix} x_{q_a} \\ y_{q_a} \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} s\cos\theta & -s\sin\theta \\ s\sin\theta & s\cos\theta \end{pmatrix} \begin{pmatrix} x_{p_i} \\ y_{p_i} \end{pmatrix},$$
$$\begin{pmatrix} x_{q_b} \\ y_{q_b} \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} s\cos\theta & -s\sin\theta \\ s\sin\theta & s\cos\theta \end{pmatrix} \begin{pmatrix} x_{p_j} \\ y_{p_j} \end{pmatrix}. \tag{2}$$

Then $q_b - q_a = G_r(p_j) - G_r(p_i)$, we have

$$\begin{pmatrix} x_{q_b} - x_{q_a} \\ y_{q_b} - y_{q_a} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} s\cos\theta & -s\sin\theta \\ s\sin\theta & s\cos\theta \end{pmatrix} \begin{pmatrix} x_{p_j} - x_{p_i} \\ y_{p_j} - y_{p_i} \end{pmatrix} \tag{3}$$

$$\Rightarrow \overrightarrow{q_aq_b} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + s \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \overrightarrow{p_ip_j} \tag{4}$$

$$\Rightarrow |\overrightarrow{q_aq_b}| = s\,|\overrightarrow{p_ip_j}|, \quad \theta_{\overrightarrow{q_aq_b}} = \theta + \theta_{\overrightarrow{p_ip_j}} \tag{5}$$

$$\Rightarrow s = \frac{|\overrightarrow{q_aq_b}|}{|\overrightarrow{p_ip_j}|}, \quad \theta = \theta_{\overrightarrow{q_aq_b}} - \theta_{\overrightarrow{p_ip_j}} \tag{6}$$

and

$$\begin{aligned} t_x &= x_{q_a} - x_{p_i}(s\cos\theta) + y_{p_i}(s\sin\theta), \\ t_y &= y_{q_a} - x_{p_i}(s\sin\theta) - y_{p_i}(s\cos\theta). \end{aligned} \tag{7}$$

In Theorem 1, (t_x, t_y, s, θ) can be determined by matching two pairs chosen in P and Q. When $|\overrightarrow{q_aq_b}|$, $|\overrightarrow{p_ip_j}|$, $\theta_{\overrightarrow{q_aq_b}}$ and $\theta_{\overrightarrow{p_ip_j}}$ are given, it needs seven multiplications "*" and five additions "+" to calculate (t_x, t_y, s, θ), i.e. one "*" for s, one "+" for θ, and six "*" and four "+" for (t_x, t_y).

From equations (3) and (4) it is clear that $\overrightarrow{p_ip_j}$ matches with $\overrightarrow{q_aq_b}$ under the new registration (0,0,s,θ). We have the following useful result.

Corollary 1. If (p_i,p_j) and (q_a, q_b) are two corresponding pairs under the registration (t_x, t_y, s, θ), then $\overrightarrow{p_ip_j}$ matches with $\overrightarrow{q_aq_b}$ under the new registration (0,0,s,θ), where $s = |\overrightarrow{q_aq_b}| / |\overrightarrow{p_ip_j}|$ and $\theta = \theta_{\overrightarrow{q_aq_b}} - \theta_{\overrightarrow{p_ip_j}}$.

Theorem 2. Let $A = \{a_1, \ldots, a_i, \ldots, a_k\}$ and $B = \{b_1, \ldots, b_i, \ldots, b_k\}$ be two sets of corresponding points under the registration $(t_{x0}, t_{y0}, s_0, \theta_0)$. Then for each corresponding pair $a_i \leftrightarrow b_i$, $A_i = \{\overrightarrow{a_ia_1}, \ldots, \vec{0}, \ldots, \overrightarrow{a_ia_k}\}$ and $B_i = \{\overrightarrow{b_ib_1}, \ldots, \vec{0}, \ldots, \overrightarrow{b_ib_k}\}$ are two sets of corresponding vectors under the new registration (0,0,s_0,θ_0).

Proof. By hypothesis, (a_i, a_j) and (b_i, b_j) are two corresponding pairs under the registration $(t_{x0}, t_{y0}, s_0, \theta_0)$ for j=1,. . .,k and j≠i. By applying Corollary 1, $\overrightarrow{a_ia_j}$ matches with $\overrightarrow{b_ib_j}$ under the new (0,0,s_0,θ_0).

In Theorem 2, the translations of two corresponding sets by each corresponding pair in them are still two corresponding sets. Because the translations in x and y directions are zero in new registration, it is not necessary to estimate them; we just need to estimate parameters s and θ.

Since $\overrightarrow{a_ia_j}$ matches $\overrightarrow{b_ib_j}$ under the new registration (0,0,s_0,θ_0) for j=1,. . .,k and j≠i, there are k−1 corresponding vectors under the new registration (0,0,s_0,θ_0) for each corresponding pair $a_i \leftrightarrow b_i$, where $s_0 = |\overrightarrow{b_ib_j}| / |\overrightarrow{a_ia_j}|$ and $\theta_0 = \theta_{\overrightarrow{b_ib_j}} - \theta_{\overrightarrow{a_ia_j}}$.

Suppose that k is the maximum number of matched pairs between P and Q under the registration $(t_{x0}, t_{y0}, s_0, \theta_0)$. If p_i and q_a are the two corresponding points, then there are other k−1 pairs of corresponding points. Therefore, there are k−1 corresponding vectors between $\overrightarrow{p_ip_j}$ under $\overrightarrow{q_aq_b}$ under the new registration (0,0,s_0,θ_0). Let the accumulator array $M_{i,a}(s, \theta)$ be used to accumulate (s,θ) determined by matching $\overrightarrow{p_ip_j}$ with $\overrightarrow{q_aq_b}$, where $s = |\overrightarrow{q_aq_b}| / |\overrightarrow{p_ip_j}|$ and $\theta = \theta_{\overrightarrow{q_aq_b}} - \theta_{\overrightarrow{p_ip_j}}$. After matching each $\overrightarrow{p_ip_j}$ with $\overrightarrow{q_aq_b}$, we discover that (s_0,θ_0) is the peak value in $M_{i,a}(s, \theta)$ and its value is k−1, where j=1,. . .,m, b=1,. . .,n, j≠i and b≠a. If p_i and q_a are not corresponding points, then there are no more k−1 corresponding vectors between $\overrightarrow{p_ip_j}$ and $\overrightarrow{q_aq_q}$. Hence, the peak value of $M_{i,a}(s, \theta)$ is less than k−1.

2.3. Determine the matching pairs support

Matching p_i in P with q_a in Q, the associated accumulator array $M_{i,a}(s, \theta)$ can be constructed. We denote the maximum value of $M_{a,i}(s, \theta)$ as w_{ia} and the associated parameters as $(s_{w_{ia}}, \theta_{w_{ia}})$, i.e. $w_{ia} = M_{i,a}(s_{w_{ia}}, \theta_{w_{ia}})$. If $w_{i,a} = c$, then it represents (a) there are other c pairs of corresponding points supporting p_i

matched with q_a, and (b) there are $c+1$ pairs of corresponding points matching under the same registration $(0, 0, s_{w_{ia}}, \theta_{w_{ia}})$. Therefore, we consider w_{ia} as *the matching pairs support* of p_i matched with q_a. The following Matching Pairs Support (MPS) finding algorithm determines $(w_{ia}, s_{w_{ia}}, \theta_{w_{ia}})$ of a given pair $p_i \leftrightarrow q_a$ by Hough Transform, where p_i in P and q_a in Q.

Matching pairs support finding algorithm

```
1. Set the accumulator array $M(s,\theta)=0$.
2. For $j=1,\ldots,m$, $j \neq i$ calculate $(|\overrightarrow{p_i p_j}|, \theta_{\overrightarrow{p_i p_j}})$.
3. For $b=1,\ldots,m$, $j \neq i$ calculate $(|\overrightarrow{q_i q_j}|, \theta_{\overrightarrow{q_i q_j}})$.
4. For $j=1,\ldots,m$, $j \neq i$, for $b=1,\ldots,n$, $b \neq a$ do
   4.1. If (match_flag[$j$][$b$]=1) do /* $p_j \leftrightarrow q_b$ is a possible matched pair. */
        4.1.1. Calculate $(s, \theta)$, where $s = |\overrightarrow{q_a q_b}| / |\overrightarrow{p_i p_j}|$ and $\theta = \theta_{\overrightarrow{q_a q_b}} - \theta_{\overrightarrow{p_i p_j}}$.
        4.1.2. Increment accumulator, $M(s, \theta) = M(s, \theta) + 1$.
5. Search the peak in $M(s, \theta)$ as $(s_{w_{ia}}, \theta_{w_{ia}})$ and $w_{ia} = M(s_{w_{ia}}, \theta_{w_{ia}})$.
```

In the MPS algorithm we introduce the match_flag array to represent the matching condition between each point in P and each point Q. If match_flag[i][a]=1, then it represents that $p_i \leftrightarrow q_a$ is a possible matching pair. If match_flag[i][a]=0, then it represents that p_i cannot match with q_a. Because we do not know which are the corresponding pairs between P and Q, each element in the match_flag array is initialized to 1.

The MPS algorithm just considers two parameters s and θ. It results in reducing the accumulating times to $(m-1)(n-1)$ and removing the effect of the variances of (t_x, t_y). Because the accumulator is a two-dimensional array, it further improves the search time to find the peak. Since the computational time of Steps 2, 3, and 5 are negligible compared to that of Step 4. The time complexity of MPS algorithm is just $(m-1)(n-1)$.

2.4. Determine the registration parameters

As $p_j \leftrightarrow q_b$ belongs to the maximum matching, w_{jb} could be the maximum number among $\{w_{ia} \mid i = 1, \ldots, m; a = 1, \ldots, n\}$. On the other hand, if w_{jb} is the maximum number among $\{w_{ia} \mid i = 1, \ldots, m; a = 1, \ldots, n\}$, then $p_j \leftrightarrow q_b$ must belong to the maximum matching, and there are other w_{jb} pairs of corresponding points supporting p_j matching with q_b. We develop the following Scaling and Rotation (SR) finding algorithm to detect which pair has *the maximum matching pairs support* and the associated registration parameters (s_0, θ_0) between two point pattern P and Q.

Scaling and rotation finding algorithm

```
1. Set pmax=0
2. Initially set match_flag[$i$][$a$]=1. /* all pairs are possible matches */
3. Assume $k=m$.
4. For $i=1,\ldots,m$, do
     For $a=1,\ldots,n$, do
       (a) Determine $(w, s_w, \theta_w)$ of pair $p_i \leftrightarrow q_a$ by using MPS algorithm.
       (b) If the support $w<$ a given support bound $t_w$, set match_flag[$i$][$a$]=0.
       (c) If pmax$<w$ set
           • (pmax,$s$,$\theta_p$)=($w$,$s_w$,$\theta_w$), pair_$i$=$i$, pair_$a$=$a$.
       (d) If pmax$\geq$ a given support $P_w$, then the support is found; go to Step 5.
       (e) If pmax$\geq m-i$, then the maximum support is found; go to Step 5.
     }
     The maximum support is not found, decrease $k$ by one, i.e. $k=k-1$.
   }
5. Pair $p_{\text{pair\_}i} \leftrightarrow q_{\text{pair\_}a}$ has the maximum matching pair support pmax and the registration parameters $(s_0,\theta_p)=(s_p,\theta_p)$.
```

Because the matching is an one-to-one mapping, the maximum number of matching pairs is not more than min(m,n). In this paper, we assume $m \leq n$ and the maximum number of matching pairs is k. Because we do not know the accurate number of matching pairs, we initially set $k=m$ in Step 3 of SR algorithm. In Step 4, for $i=1$ we begin to match p_1 in P with each point q_a in Q and then search the maximum pairs support pmax among $\{w_{1a} \mid a = 1, \ldots, n\}$. If pmax$=m-1$, then there are m corresponding pairs between P and Q. Hence, the maximum support is found. If pmax is less than $m-1$, then the maximum number of matching pairs must be less than m; otherwise, pmax must be equal to $m-1$. Decrease k by one and continue to search the maximum among next support $\{w_{2a} \mid a = 1, \ldots, n\}$.

Now if the maximum number of matching pairs is $(m-c+1)$, then a p_i in $\{p_1, p_2, \ldots, p_c\}$ exists such that the pmax equal to $m-c$, where pmax is the maximum among $\{w_{ia} \mid i = 1, \ldots, c; a = 1, \ldots, n\}$. If pmax$=m-c$, then the maximum matching pairs support is found; if pmax$<m-c$, then the maximum number of matching pairs must be less than $(m-c+1)$. Decrease k by one and continue to search other supports. The inference rules are stated as follows:

$$
\begin{array}{l}
k = m. \\
\text{For } i = 1, \text{ if pmax} < m - 1, \text{ it means } k \leq m - 1. \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{Let } k = k - 1. \\
\text{For } i = 2, \text{ if pmax} < m - 2, \text{ it means } k \leq m - 2. \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{Let } k = k - 1. \\
\qquad\qquad\qquad\qquad \vdots \\
\text{For } i = c, \text{ if pmax} < m - c, \text{ it means } k \leq m - c. \\
\qquad\qquad\qquad\qquad\qquad\qquad \text{Let } k = k - 1.
\end{array}
\tag{8}
$$

Hence, for $i=c$, if pmax$\geq m-c$, it means the maximum pairs support is found.

If we find a matching which has at least P_w+1 matching pairs between P and Q, we just need to seek the pairs' support which has the value P_w, where P_w is a given support. This statement is shown in Step 4(d).

In Step 4(b), if the support of $p_i \leftrightarrow q_a$ is less than a given bound t_w, then $p_i \leftrightarrow q_a$ does not belong to a matching which at least has the t_w+1 matching pairs; we set the match_flag[i][a]=0.

If the maximum number of matching pairs is k, the SR algorithm just seeks which is the maximum support among $\{w_{ia} \mid i = 1, \ldots, m - k + 1; a = 1, \ldots, n\}$. Hence, the time complexity of SR algorithm is $c \cdot n \cdot$(time complexity of MPR algorithm)$=c \cdot n \cdot (m-1)(n-1)$, where c is constant and c=m−k+1.

3. DETERMINATION OF MATCHING PAIRS

In Section 2, we estimate two corresponding points which have *the maximum matching pairs support* and the makes there are maximum matching pairs under the registration parameters (s_0,θ_0). The purpose of this section is to determine these matching pairs between P and Q. The following Matching Pairs Determination (MPD) algorithm which determine these matching pairs between P and Q is based on the assumption that two corresponding points and the registration parameter (s_0,θ_0) are known.

Matching pairs determination algorithm

```
1. For $j$=1,. . .,$m$, $j \neq i$ do
   (a) Set $ib$=0, count=0.
   (b) For $b$=1,. . .,$n$, $b \neq a$ do
       • If (flag[$j$][$b$]=1) do
         • Calculate $s = \mid \overrightarrow{q_a q_b} \mid / \mid \overrightarrow{P_i P_j} \mid$ and $\theta = \theta_{\overrightarrow{q_a q_b}} - \theta_{\overrightarrow{p_i p_j}}$.
         • If $\{\mid s - s_0 \mid \leq \Delta s$ and $\mid \theta - \theta_0 \mid \leq \Delta\theta\}$ do
           • $ib$=$b$, count=count+1.
           • Collect this pair $p_j \leftrightarrow q_b$ as a possible matching pair into set $F$.
   (c) If (count=1), then the pair $p_j \leftrightarrow q_{ib}$ is taken as a matching pair. And collect it into set $G$.
2. Estimate the first optimal registration parameters $r_1 = (t_{x_1}, t_{y_1}, s_1, \theta_1)$ between those matching pairs in set $G$.
3. For each $p_j \leftrightarrow q_b$ in $F$, select $q_b$ which is the closet point to $G_{r_1}(p_j)$ and $\|G_{r_1}(p_j) - q_b\| \leq d_1$. The pair $p_j \leftrightarrow q_b$ is taken as the matching pair.
4. For each $p_j$ in $P$, select which $q_b$ is the closet point to $G_{r_1}(p_j)$ and $\|G_{r_1}(p_j) - q_b\| \leq d_1$. The $p_j \leftrightarrow q_b$ is taken as the matching pair.
5. Estimate the optimal registration parameters $(t_x, t_y, s, \theta)$ between these matching pairs.
```

Without the loss of generality, we suppose that the estimated corresponding points are p_i and q_a. Due to the quantization of parameter space, the estimated registration parameters (s_0,θ_0) are just discrete values. The (s_{jb},θ_{jb}) is computed after matching p_j in P with each q_b in Q, where $s_{jb} = \mid q_a q_b{}^{\rightarrow} \mid / \mid p_i p_j{}^{\rightarrow} \mid$ and $\theta_{jb} = \theta_{q_a q_b{}^{\rightarrow}} - \theta_{p_i q_j{}^{\rightarrow}}$.

If p_j and q_b are corresponding points, then (s_{jb},θ_{jb}) would be very close to (s_0,θ_0). Hence, for point p_j, if there is only one (s_{jb},θ_{jb}) within the region bound of (s_0,θ_0), the p_j and q_b are a matching pair. Collect this matching pair into set G. This statement is shown in Step 1(c). If there are more than one (s_{jb},θ_{jb}) within the region bound of (s_0,θ_0), then we do not know which q_b is the corresponding point of p_j and we take these point pairs as possible matching pairs. Collect these possible matching pairs into set F. This statement is shown in Step 1(b).

The region bound of (s_0,θ_0) could be squares, circles or polygons. In this paper, the region bound of (s_0,θ_0) is given by $s_0 \pm \Delta s$ and $\theta_0 \pm \Delta\theta$, where Δs is the allowable scaling bound and $\Delta\theta$ is the allowable rotation bound. Here we note that Δs and $\Delta\theta$ are just used to choose possible corresponding pairs.

In Step 2, we use the formula in Appendix A to estimate the first optimal registration parameters $(t_{x_1}, t_{y_1}, s_1, \theta_1)$ between those matching pairs in set G and let $r_1 = (t_{x_1}, t_{y_1}, s_1, \theta_1)$ in Step 2.

After the first optimal registration parameters are estimated, we transform set P into another by these parameters. For each $p_j \leftrightarrow q_b$ in F, we check which q_b is the closest point to $G_{r_1}(p_j)$ and check the coordinates differences within the distortion bound d_1, i.e. $\|G_{r_1}(p_j) - q_b\| \leq d_1$. This closest pair is taken as the matching pair.

If noise and distortion are larger and the values of Δs and $\Delta\theta$ are given too small, then some matching pairs may not be chosen as the possible matching pairs. The problem can be solved by the following method. For each p_j in P, we check which q_b in Q is the closest point to $G_{r_1}(p_j)$ and check the coordinates differences within the distortion bound d_1, i.e. $\|G_{r_1}(p_j) - q_b\| \leq d_1$. This closest pair is taken as the matching pair.

After collecting these matching pairs, we use the formula in Appendix A to estimate the optimal registration parameters (t_x, t_y, s, θ).

4. EXPERIMENTS

In this section, several experiments are carried out on both synthetic and real data to measure the performance of the proposed point pattern matching algorithm.

For the synthetic data, two point patterns P and Q are generated under the circumstances that there are k corresponding points between them. The data sets are generated as suggested in references (18,27). We run a large number of simulations by using randomly generated data. In each trial, P is a set of m random points chosen from a two-dimensional uniform distribution over (0.0,256.0)×(0.0,256.0), i.e. points in P are randomly chosen from 256×256 images, and the distance between two points in P is larger than 10.0, i.e. $\|p_i - p_j\| > 10$ (Fig. 1). Q is the transformation of P by $G_{(t_x, t_y, s, \theta)}$ as follows:

$$\begin{pmatrix} x_{q_i} \\ y_{q_i} \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} s\cos\theta & -s\sin\theta \\ s\sin\theta & s\cos\theta \end{pmatrix} \begin{pmatrix} x_{p_i} \\ y_{p_i} \end{pmatrix}, \quad i = 1, \ldots, m,$$

where s=1.30, θ=40°, t_x=250.0, and t_y=0.0, respectively; the index of Q is reverse order, i.e.

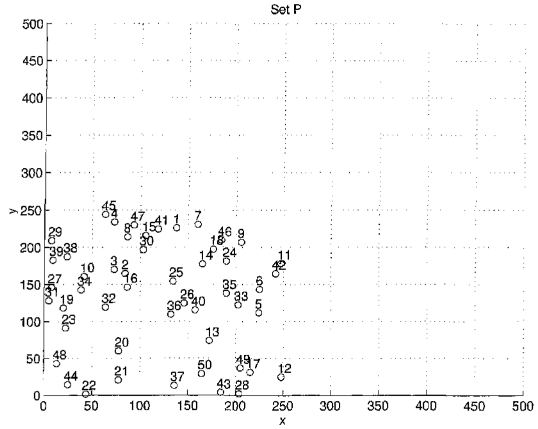

Fig. 1. P is a set of 50 random points chosen from a two-dimensional uniform distribution over (0.0,256.0)×(0.0,256.0).

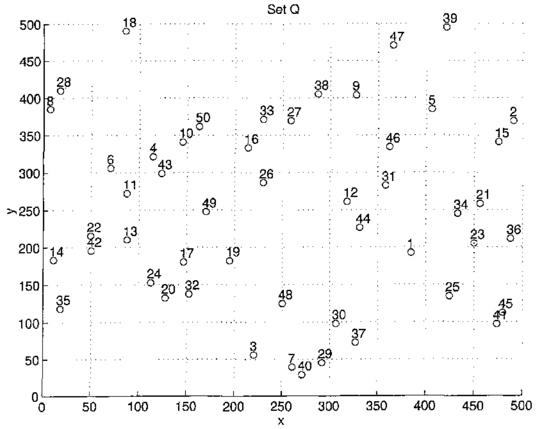

Fig. 2. Q is the transformation of P by $G_{(t_x,t_y,s,\theta)}$ and the index of Q is reverse order, i.e. $q_i = (t_x, t_y, s, \theta) = (250.0, 0, 1.3, 40^\circ)$ and m=50.

$q_i = G_r(p_{m-i+1})$, $r = (t_x, t_y, s, \theta)$, i=1,...,m (Fig. 2). The distortion condition is simulated by adding noise to points in Q, where the noise matching condition is simulated by deleting d point from Q and then randomly inserting d points in Q and we get (m−d) matching pairs between P and Q, i.e. k=(m−d).

Here m and n are chosen as 50. The allowable scaling bound Δs is 0.08 and the allowable rotation bound $\Delta\theta$ is 3°. The distortion bound d_1 is 6.0. Simulations are made with various values of d and p. One hundred trials are tested for each case. A trial is accepted as a success if k correct matching pairs are detected between P and Q. The success ratio is computed as the ratio of the number of successful trials to the total number trials. The average execution time (user time) is computed as the ratio of the total execution times to the total number trials on Sun-4 workstation.

The results of all cases are summarized in Table 1. It shows that the performance of proposed algorithm is quite well. Each case has a success rate of 100%. Even as two point sets which have 60% corresponding points between them and local distortion with noise generated uniformly over (0.0,5.0)×(0.0,5.0) added to all points of one set, row 12 of Table 1 shows correct matching results for all trials. The average execution times are about 0.5 s. The cost increases as the different points between them increase.

To test on real data, two fingerprints captured from the same finger are used for registration. The images are shown in Fig. 3. After preprocessing process, the extracted feature points are shown in Fig. 4. These feature points are the points of ridge endings (terminations) or bifurcations (branchings). In Fig. 4 left figure has 20 endings and 29 branchings and right figure has 31 endings and 14 branchings. Thus m=49 and n=45. After we apply the proposed algorithm to match left figure with right figure in Fig. 4, 34 matched pairs were detected. Figure 5 shows these matched pairs which are marked with "big black dots". The optimal parameters $r = (t_x, t_y, s, \theta) = (56.6, -44.4, 1.02, 16.3^\circ)$ are obtained. The mean squared error is 4.73. Observing Figs 3 and 5, we can confirm that the estimated matched pairs are true matched pairs and the rotation angle (16.3°) is correct.

Based on the results, it is shown that the proposed matching algorithm is invariant to translations, rotations, scale changes, local distortion, and extra or missing points conditions.

Table 1. Accuracy and speed of the proposed point pattern matching algorithm for all cases

Rows	m=n	Matched pairs k	Noise level p	Success (%)	Average (s)
1	50	50	0	100	0.32
2	50	50	5	100	0.32
3	50	45	0	100	0.34
4	50	45	5	100	0.34
5	50	40	0	100	0.39
6	50	40	5	100	0.39
7	50	35	0	100	0.43
8	50	35	5	100	0.43
9	50	30	0	100	0.46
10	50	30	5	100	0.47

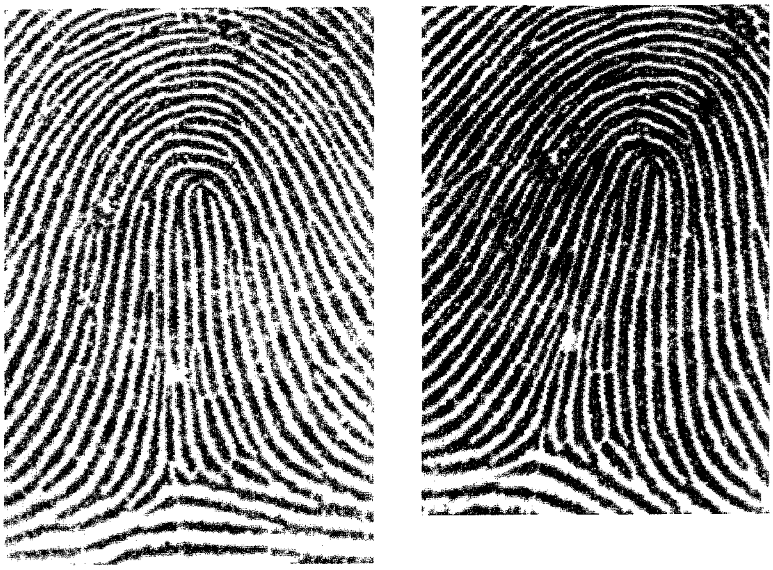

Fig. 3. Two fingerprints are captured from the same finger.

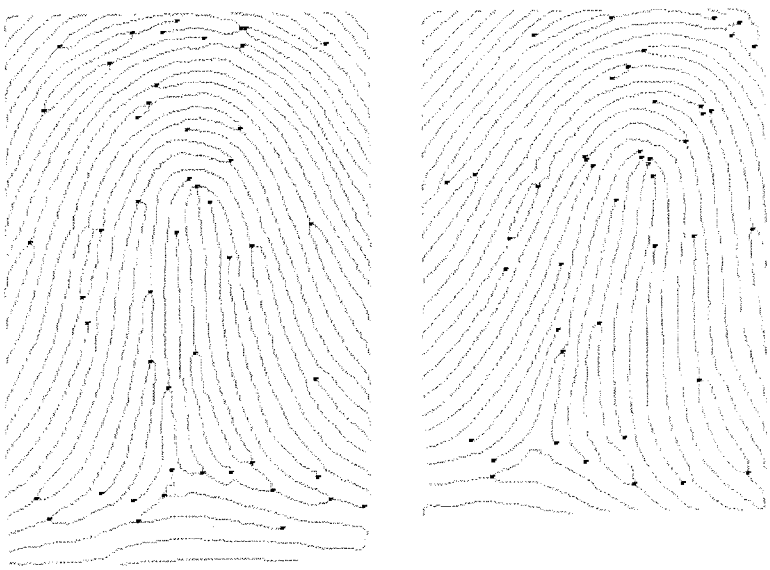

Fig. 4. The feature points of Fig. 3. Left figure has 20 end points and branch points. Right figure has 31 end points and 14 branch points.

5. CONCLUSION

In this paper we have presented a novel point matching algorithm that is invariant to arbitrary translations, rotations, scaling changes, local distortion, and extra/missing points conditions. Besides, theories and method have been developed to determine the registration (scaling and rotation) of a given point pair that makes the most of point pairs match under the registration.

In our cluster approach, the dimension of the parameter space is reduced from four (scaling s, rotation θ, x translation t_x and y translation t_y) to two (scaling and rotation). There are several advantages to do this. First, it removes the noise effects of translations (t_x,t_y) so that the algorithm is more robust. For two point sets which have 60% corresponding points between them and local distortion with noise generated uniformly over $(0.0,5.0)\times(0.0,5.0)$ added to all points of one set, the success rate of matching is still 100%. Second, the algorithm can match under arbitrary translations because the (t_x,t_y) is not considered here. Although the translations along the x and y directions is 250.0 and 0.0 in the synthetic data, the success rate of matching is still 100%. Third, it requires small memories during the cluster process and improves the search time to find the peak in the accumulator array because the accumulator is just a two-dimensional array. Fourth, the algorithm is fast and efficient because the 2D cluster approach is simple and quick. Each accumulating time just needs

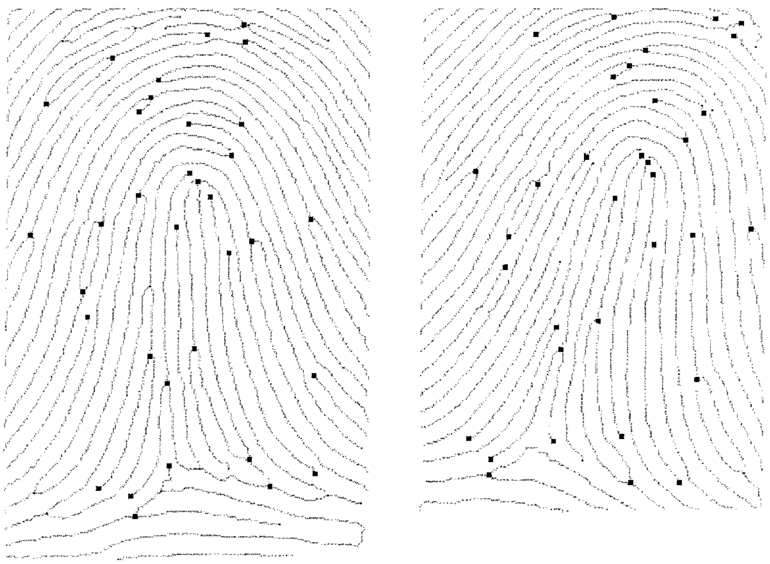

Fig. 5. Matching result of Fig. 4, where m=49, n=45, matching pairs=34, $r = (t_x, t_y, s, \theta) = (56.6, -44.4, 1.02, 16.3°)$, and mean squared error is 4.73.

one multiplication and one addition to calculate (s,θ), and it need not compute (t_x,t_y) during the cluster process. For 50 to 50 points matching, the average execution time is about 0.5 s.

The proposed point matching algorithm can be expanded to line pattern matching. Although the algorithm is invariant to 2D transformations, it can also be applied to 3D transformations.

APPENDIX

In this appendix, we show a formula which gives the least-squares estimation of registration parameters between two corresponding point subsets. If k pairs of corresponding points $\{a_i \leftrightarrow b_i \mid i = 1, \ldots, k\}$ between two sets P and Q are available ($k \geq 2$), we want to find the transformation parameters (t_x,t_y,s,θ) given the minimum value of the sum of squared coordinates differences $S(t_x,t_y,s,\theta)$ of these corresponding points. $S(t_x, t_y, s, \theta) = \sum_{i=1}^{k} e_i^{\mathrm{T}} e_i$, where e_i is the coordinates difference between b_i and the transformed point $G(a_i)$,

$$\begin{aligned} e_i &= \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} s\cos\theta & -s\sin\theta \\ s\sin\theta & s\cos\theta \end{pmatrix} \begin{pmatrix} x_{a_i} \\ y_{a_i} \end{pmatrix} - \begin{pmatrix} x_{b_i} \\ y_{b_i} \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & x_{a_i} & -y_{a_i} \\ 0 & 1 & y_{a_i} & x_{a_i} \end{pmatrix} \begin{pmatrix} t_x \\ t_y \\ s\cos\theta \\ s\sin\theta \end{pmatrix} - \begin{pmatrix} x_{b_i} \\ y_{b_i} \end{pmatrix} \\ &= C_{a_i} r - b_i, \end{aligned} \qquad \text{(A1)}$$

where $r = (t_x, t_y, s\cos\theta, s\sin\theta)^{\mathrm{T}}$and

$$C_{a_i} = \begin{pmatrix} 1 & 0 & x_{a_i} & -y_{a_i} \\ 0 & 1 & y_{a_i} & x_{a_i} \end{pmatrix}.$$

Collecting these k coordinates differences, we have

$$\begin{aligned} E &= \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_k \end{pmatrix} = \begin{pmatrix} C_{a_1} r - b_1 \\ C_{a_2} r - b_2 \\ \vdots \\ C_{a_k} r - b_k \end{pmatrix} = \begin{pmatrix} C_{a_1} \\ C_{a_2} \\ \vdots \\ C_{a_k} \end{pmatrix} r - \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{pmatrix} \\ &= C \cdot r - b, \end{aligned} \qquad \text{(A2)}$$

where $b^{\mathrm{T}} = (b_1^{\mathrm{T}}, b_2^{\mathrm{T}}, \ldots, b_k^{\mathrm{T}})$ and $C^{\mathrm{T}} = (C_{a_1}^{\mathrm{T}}, C_{a_2}^{\mathrm{T}}, \ldots, C_{a_k}^{\mathrm{T}})$.

Rewriting S, we express S as below:

$$\begin{aligned} S(r) &= \sum_{i=1}^{k} e_i^{\mathrm{T}} e_i = (e_1^{\mathrm{T}} e_2^{\mathrm{T}} \cdots e_k^{\mathrm{T}}) \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_k \end{pmatrix} \\ &= E^{\mathrm{T}} E = (Cr - b)^{\mathrm{T}} (Cr - b) \\ &= r^{\mathrm{T}} C^{\mathrm{T}} Cr - b^{\mathrm{T}} Cr - r^{\mathrm{T}} C^{\mathrm{T}} b + b^{\mathrm{T}} b \\ &\quad (\text{because } r^{\mathrm{T}} C^{\mathrm{T}} b = b^{\mathrm{T}} Cr) \\ &= r^{\mathrm{T}} C^{\mathrm{T}} Cr - 2b^{\mathrm{T}} Cr + b^{\mathrm{T}} b. \end{aligned} \qquad \text{(A3)}$$

Now, our objective is to determine the $\hat{r}$ that minimizes $S(r)$. We take the vector derivative of $S(r)$ with respect to r; but before doing this, we recall the following from vector calculus:

If b and g are two $n \times 1$ nonzero vectors, and Q is an $n \times n$ symmetric matrix, then

$$\frac{\mathrm{d}}{\mathrm{d}g}(b^{\mathrm{T}} g) = b \qquad \text{(A4)}$$

and

$$\frac{\mathrm{d}}{\mathrm{d}g}(g^{\mathrm{T}} Q g) = 2Qg. \qquad \text{(A5)}$$

Using these formulas, we find that

$$\frac{\mathrm{d}S(r)}{\mathrm{d}r} = 2C^{\mathrm{T}}Cr - 2C^{\mathrm{T}}b. \qquad \text{(A6)}$$

Set d$S(r)$/dr=0. We obtain the following formula for $\hat{r}$,

$$\hat{r} = [C^{\mathrm{T}}C]^{-1}C^{\mathrm{T}}b. \qquad \text{(A7)}$$

For equation (A7) to be valid, C^TC must be invertible. This is always true when C is of maximum rank, i.e. rank(C)=4. If k=2, we have

$$C = \begin{pmatrix} 1 & 0 & x_{a_1} & -y_{a_1} \\ 0 & 1 & y_{a_1} & x_{a_1} \\ 1 & 0 & x_{a_2} & -y_{a_2} \\ 0 & 1 & y_{a_2} & x_{a_2} \end{pmatrix} \Rightarrow \mathrm{rank}(C) = 4. \qquad \text{(A8)}$$

Therefore, if $k \geq 2$ then equation (A7) is valid and unique. How do we know that $\hat{r}$ minimizes $S(r)$? We compute d$^2S(r)$/dr^2 and see if it is positive define (i.e. $\hat{r}$ minimizes $S(r)$ if d$S(r)$/dr=0 and d$^2S(r)$/dr^2 is positive). Doing this, we see that

$$\frac{\mathrm{d}^2S(r)}{\mathrm{d}r^2} = 2C^{\mathrm{T}}C > 0 \qquad \text{(A9)}$$

so $\hat{r}$ can minimize $S(r)$, and

$$\hat{r} = [C^TC]^{-1}C^Tb$$
$$= \frac{1}{\det}\begin{pmatrix} l_A & 0 & -\mu_{x_A} & \mu_{y_A} \\ 0 & l_A & -\mu_{y_A} & -\mu_{x_A} \\ -\mu_{x_A} & -\mu_{y_A} & k & 0 \\ \mu_{y_A} & -\mu_{x_A} & 0 & k \end{pmatrix}\begin{pmatrix} \mu_{x_B} \\ \mu_{y_B} \\ l_{A+B} \\ l_{A-B} \end{pmatrix}, \qquad \text{(A10)}$$

where

$$\mu_{x_A} = \sum_i^k x_{a_i}, \quad \mu_{x_B} = \sum_i^k x_{b_i}, \qquad \text{(A11)}$$

$$\mu_{y_A} = \sum_i^k y_{a_i}, \quad \mu_{y_B} = \sum_i^k y_{b_i}, \qquad \text{(A12)}$$

$$l_{A+B} = \sum_i^k (x_{a_i}x_{b_i} + y_{a_i}y_{b_i}), \qquad \text{(A13)}$$

$$l_{A-B} = \sum_i^k (x_{a_i}y_{b_i} - y_{a_i}x_{b_i}), \qquad \text{(A14)}$$

$$l_A = \sum_i^k (x_{a_i}^2 + y_{a_i}^2) \qquad \text{(A15)}$$

and

$$\det = k * l_A - \mu_{x_A}^2 - \mu_{y_A}^2. \qquad \text{(A16)}$$

REFERENCES

1. P. C. Yuen, P. W. M. Tsang and F. K. Lam, Robust matching process: a dominant point approach, *Pattern Recognition Lett.* **15**, 1223–1233 (December 1994).
2. X. Hu and N. Ahuja, Matching point features with ordered geometric, rigidity and disparity constraints, *IEEE Trans. Pattern Analysis Mach. Intell.* **16**(10), 1041–1049 (October 1994).
3. P. W. M. Tsang, P. C. Yuen and F. K. Lam, Classification of partially occluded objects using 3-point matching and distance transformation, *Pattern Recognition* **27**(1), 27–40 (1994).
4. M. C. K. Yang and Jong-Sen Lee, Object identification from multiple images based on point matching under a general transformation, *IEEE Trans. Pattern Analysis Mach. Intell.* **16**(7), 751–756 (July 1994).
5. S. Umeyama, Parameterized point pattern matching and is application to recognition of object families, *IEEE Trans. Pattern Analysis Mach. Intell.* **15**(2), 136–144 (February 1993).
6. L. G. Brown, A survey of image registration techniques, *ACM Computing Surveys* **24**(4), 325–376 (December 1992).
7. H. P. Trivedi, Semi-analytic method for estimating stereo camera geometry from matched points, *Image Vision Comput.* **9**(2), 75–81 (April 1991).
8. H. P. Trivedi, A semi-analytic method of determining stereo camera geometry from matched points in a pair of images: Coincident meridional planes, exact or noisy data, *Computer Vision Graphics Image Process.* **51**, 299–312 (1990).
9. Min-Hong Han and D. Jang, The use of maximum curvature points for the recognition of partially occluded objects, *Pattern Recognition* **1/2**, 21–33 (1990).
10. O. D. Faugeras and S. Maybank, Motion from point matches: Multiplicity of solutions, *Int. J. Comput. Vision* **4**, 225–246 (1990).
11. R. M. Haralick, C. N. Lee, X. Zhuang, V. G. Vaidya and M. B. Kim, Pose estimation from corresponding point data, *Proc. IEEE Computer Society Workshop on Computer Vision*, 258–263 (1987).
12. Cheng-Ye Wang, H. Sun, S. Yada and A. Rosenfeld, Some experiments in relaxation image matching using corner features, *Pattern Recognition* **16**(2), 167–182 (1983).
13. J. Sprinkaz and M. Werman, Affine point matching, *Pattern Recognition Lett.* **15**, 337–339 (April 1994).
14. S. Umeyama, Least-squares estimation of transformation parameters between two point patterns, *IEEE Trans. Pattern Analysis Mach. Intell.* **13**(4), 376–380 (April 1991).
15. D. Lavine, B. A. Lambaird and L. N. Kanal, Recognition of spatial point patterns, *Pattern Recognition* **16**, 289–295 (1983).
16. H. Ogawa, Labeled point pattern matching by delaunay triangulation and maximal cliques, *Pattern Recognition* **19**(1), 35–40 (1986).
17. J. Hong and X. Tan, A new approach to point pattern matching, *Proc. 9th Int. Conf. Pattern Recognition*, 82–84 (1988).
18. V. V. Vinod and S. Ghose, Point matching using asymmetric neural networks, *Pattern Recognition* **26**(8), 1207–1214 (1993).
19. S. Ranade and A. Rosenfeld, Point pattern matching by relaxation, *Pattern Recognition* **12**, 269–275 (1980).
20. J. Ton and A. K. Jain, Registering landsat images by point matching, *IEEE Trans. Geosci. Remote Sensing* **27**(5), 642–651 (September 1989).
21. S. Z. Li, Matching: Invariant to translations, rotations and scale changes, *Pattern Recognition* **25**(6), 583–594 (1992).
22. Li-Der Jeng, Chih-Jen Huang, Wen-Hsing Hsu and Fang-Hsuan Cheng, Planar pattern matching based on minimum feature relations, in *Proc. 1993 Int. Computer Symp. Conf.*, 885–892 (1993).
23. R. S. Mitra and N. N. Murthy, Elastic, maximal matching, *Pattern Recognition* **24**(8), 747–753 (1991).
24. H. Ogawa, Labeled point pattern matching by fuzzy relaxation, *Pattern Recognition* **17**(5), 569–573 (1984).
25. D. Skea, Barrodale, R. Kuwahara and R. Poeckert, A control matching algorithm, *Pattern Recognition* **26**(2), 269–276 (1993) .

26. G. Stockman, S. Kopstein and S. Benett, Matching images to models for registration and object detection via clustering, *IEEE Trans. Pattern Analysis Mach. Intell.* **PAMI-4**(3), 229–241 (May 1992).

27. A. Goshtasby and G. C. Stockman, Point pattern matching using convex hull edges, *IEEE Trans. Systems Man Cybernet.* **SMC-15**(5), 631–637 (September/October 1985).

About the Author — SHIH-HSU CHANG was born in Changhua, Taiwan, on 27 January 1965. In 1987, he received the B.E. in control engineering from National Chiao Tung University, Taiwan. He is now a research student in the Department of Electrical Engineering, National Tsing Hua University, Taiwan. His current interests include computer vision, image processing and pattern recognition. He is a member of the IEEE Computer Society and IPPR of R.O.C.

About the Author — FANG-HSUAN CHENG was born in Hsinchu, Taiwan, on 13 June 1960. In 1982, he graduated with a B.E. degree from the Department of Electrical Engineering, National Cheng-Kung University, Taiwan. He received the M.E. and Doctor of Engineering degrees from the Institute of Electrical Engineering, National Tsing Hua University, Taiwan, in 1984 and 1988, respectively. He got an honor of Dragon Totem Award from Acer Corporation in 1988. From 1988 to 1992, Dr Cheng was with Chung Shan Institute of Science and Technology as a senior specialist. There, he was involved in signal processing, flight data analysis, parameter estimation and distributed database design. Dr Cheng has been an Associate Professor in the Department of Information and Computer Engineering at Chung Yuan Christian University in Taiwan since 1991. In the fall of 1992, Dr Cheng joined the Department of Computer Science at Chung Hua Polytechnic Institute, where he is currently an associate professor. Prof. Cheng is also one of the members of editorial board of Communications of COLIPS, an international journal of Singapore. His current research interests include signal processing, coding and information theory, color image processing, 3D modeling, multimedia, Chinese OCR etc.

About the Author — WEN-HSING HSU was born in Taipei, on 17 May 1950. He received a B.S. in electrical engineering from National Cheng Kung University in 1972, M.E. and Doctor of Engineering degrees in Electrical Engineering from Keio University, Tokyo, Japan, in 1978 and 1982, respectively. In 1982, he joined the Department of Electrical Engineering at National Tsing Hua University, and is currently a Professor there. He also joined the Institute of Information Science as an associate research fellow in 1984 and is currently a research fellow there. Since August 1990, he has been the Deputy General Director of Opto-Electronics and Systems Lab., Industrial Technology Research Institute. He received a Technology Discovery Award from the Sun Yat-Sen Cultural Foundation in 1986, and was honored as an Outstanding Person on Information Science in the Republic of China in 1986, and one of the ten outstanding engineers in the Republic of China by the Chinese Institute of Engineers in 1988. He also received an Outstanding Research Award from the National Science Council, Republic of China in 1988, 1989 and 1990 and a Best Paper Award from the Chinese Institute of Engineers in 1989. His research interests include image processing, pattern recognition, parallel algorithm, neural computer and fuzzy theory. Dr Hsu is a member of the Institute of Electronics, Information and Communication Engineers (IEEE), Information Processing Society of Japan, and IEEE Computer Society of USA.

About the Author — GUO-ZUA WU was born in Taiwan, in May 1961. He received the B.E. and M.E. degrees in mechanical engineering from National Chiao Tung University, Taiwan, in 1983 and 1985, respectively. He has been a electrical engineer and project leader of OES/ITRI since 1985. He is also a Ph.D. graduate student in the Department of Electrical Engineering, National Tsing Hua University, Taiwan. His current interests include pattern recognition, digital signal processing, control theory and image processing.