

GA-Based Affine PPM Using Matrix Polar Decomposition

Mehdi Ezoji

¹Electrical Engineering
Department, Amirkabir
University of Technol-
ogy, Tehran, Iran
m_ezoji@aut.ac.ir

Karim Faez

Electrical Engineering
Department, Amirkabir
University of Technol-
ogy, Tehran, Iran
kfaez@aut.ac.ir

Majid Ziaratban

Electrical Engineering
Department, Amirkabir
University of Technol-
ogy, Tehran, Iran
m_ziaratban@aut.ac.ir

Saeed Mozaffari^{1,2}

²Electrical and Com-
puter Department,
Qazvin Azad Univer-
sity, Iran
s_mozaffari@aut.ac.ir

Abstract

Point pattern matching (PPM) is an important problem in pattern recognition, digital video processing and computer vision. In this paper, novel and fast procedure based on Genetic Algorithm, for affine PPM is described.

Most matching techniques solved the PPM problem by determining the correspondence between points localized spatially within two sets, then to get the proper transformation parameters, solved a set of equations. In this paper, we use this fact that correspondence and transformation matrices are two unitary polar factors of Grammian matrices. We estimate one of this factors by the Genetic Algorithm's population and evaluate this estimation by computing another factor using fitness function. This approach is easily implemented one and because of using the genetic algorithm, don't converge into local minima. Simulation results on randomly generated points patterns and real point patterns, show that the algorithm is very effective.

1 Introduction

The Point pattern matching (PPM) is an important problem in pattern recognition, digital video processing and computer vision. There are many applications such as object recognition, motion tracking, image registration, stereo matching and pose estimation that require so called PPM. Usually, most matching algorithms first determine the point-wise feature correspondence between two patterns, then establish a set of equations that describe a transformation with unknown parameters. To get the answers, the equation set must be solved. Although this scheme seem to be simple, because of an interlock relationship between feature correspondence and transformation estimation, solving this problem is a trivial task [1][2][3].

Currently, a large number of different approaches are proposed on PPM. Goshtasby et al.[4] reduced the complexity of the matching problems using the convex hulls of the point sets on the assumption that there is no spurious or lost points on the convex hulls. Ballard et al. [5] used generalized hough transform to detect arbitrary patterns. Stockman et al.[6] estimated the similarity transformation parameters using pose clustering, then determined the correspondence matrix. Steven Gold et al.[7] attempted to find the pose and correspondence simultaneously in a two stage iterative algorithm which is derived by minimizing an energy function describing the problem. Their algorithm works for 2-D and 3-D PPM. Zhang et

al.[8] presented a method based on genetic algorithm to solve the PPM problem under general affine transformation using partial bidirectional Hausdorff distance. Marco Carcassoni et al. [3] used spectral graph analysis for PPM. They focused on the use of property matrix spectra for the correspondence matching. Van Wamelen et al.[9] presented a fast expected time algorithm for 2-D PPM, using a probabilistic technique.

More recently, several researchers have focused on description of the PPM problem, by an objective function which must be minimized to solve the problem [7].

The steps of our method are shown in Fig. 1. The outline of this paper is as follows: In Section 2, we define the problem. In Section 3, we provide theoretical prerequisites of our method. In Section 4, we state principal of our algorithm and in Section 5, we provide details of how to implements our algorithm. Finally, we demonstrate the effectiveness of our method on real images and on the synthetic sets, in section 6.

2 Statement of the problem

Given two point pattern sets, called the Model and the Scene. These sets contain feature points as n-tuple vectors. $M = \{M_1, M_2, \dots, M_m\}$ and $S = \{S_1, S_2, \dots, S_s\}$. In particular, m is smaller than or equal to s. Does object which described by the Model, occur in the Scene?

To formulate this problem, we construct two matrices, calling the first one M and the other S as follows :

$$M = \begin{bmatrix} M_{11} & \dots & M_{1n} \\ \vdots & \dots & \vdots \\ M_{m1} & \dots & M_{mn} \end{bmatrix}_{m \times n} \quad \text{and} \quad S = \begin{bmatrix} S_{11} & \dots & S_{1n} \\ \vdots & \dots & \vdots \\ S_{s1} & \dots & S_{sn} \end{bmatrix}_{s \times n} \quad (1)$$

Notice that we don't know which row of S corresponds to which row of M. To state this problem, we define the correspondence matrix C as:

$$C_{ij} = \begin{cases} 1 & S_i \text{ corresponds to } M_j \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

If there exist a geometric transformation T between the two matrices M and S, our problem changes into finding the matrix C and T such that:

$$S = CMT \quad (3)$$

There are two unknown matrices in the equation [3] such that to find one of them, we require knowledge about the other. Our strategy to solve this problem is to

find C and T simultaneously.

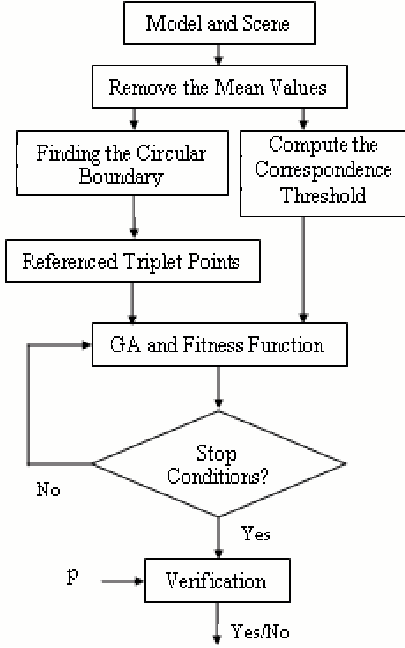


Figure 1. Block diagram of Proposed Algorithm

3 Principle of proposed algorithm

In order to describe our method, we separate it into different parts and analyse each part theoretically.

3.1 PPM and matrix polar decomposition

Definition: The polar decomposition of a full rank rectangular matrix $G \in R^{m \times n}$, is defined by :

$$G = C_{m \times n} M_{n \times n} \quad \text{where} \quad M = M^T, \quad C^T C = I \quad (4)$$

Constructing two Grammian matrices[1] using the model and scene matrices, we have:

$$\begin{aligned} G_1 &= S M^T \\ G_2 &= M^T S \end{aligned} \quad (5)$$

Hongdong Li et al.[1][2], show that if we estimate the transformation matrix T roughly and compensate it, the best correspondence matrix C is the unitary polar factor of G_1 . Because

$$\begin{aligned} M_{compensated} &= M T \\ G_1 &= C M_{compensated} M_{compensated}^T \end{aligned} \quad (6)$$

Therefore, finding C matrix is equivalent to the following minimization problem:

$$\min_C \left\| C - G_1 \right\|_{2-norm}^{ideal} \quad (7)$$

They also show that, if C has been estimated roughly, the best transformation matrix T is unitary polar factor of G_2 in ideal case. Similarly, finding T matrix is equivalent to the following minimizing problem:

$$\min_R \left\| T - G_2 \right\|_2^{ideal} \quad (8)$$

In this work, we assume that the similarity transformation T is a affine transformation. An affine transformation is any transformation that preserves collinearity and ratios of distances. A affine transformation can be transform a rectangular to a parallelogram, it does not necessarily preserve angles or lengths. Any triangle can be transformed into any other by an affine transformation.

4 An overview of proposed method

According to the previous section, there are a duality relationship between C and T matrices. Therefore, our method to solve the PPM problem is as follows:

Firstly, we remove the mean value for two sets, then obtain the smallest circles which encloses them, named C_m and C_s . We select a triple of points on the C_m that construct an equilateral triangle (these points may not be the real points of the model set) as the referenced points, then we generate a population of N individuals which are triple of points located around the C_s , then we determine the transformation matrices $T_k, k=1:N$ that maps referenced points to the kth individual. For each individual, we compensate the model set by multiplying M by T_k and then obtain the correspondence matrix C_k by considering the correspondence threshold (cth). In this phase, each $A \in (M T_k)$ is paired with $B \in Scene$ if $\|A - B\| \leq cth$. We use the correspondence threshold for increasing the robustness of our algorithm to noise (Computing the reasonable value for the correspondence threshold will be described later). To evaluate the T_k matrix, we define an objective function as follows:

$$f(C_k) = \text{norm}(C_k - G_1^{compensated})_2 \quad (9)$$

$f(C_k)$ can be interpreted as the mismatching between two point sets according to T_k . Therefore, we define fitness function by $f(C_k)$. The output of GA is the triple of points which returns the lowest value of mismatching. Finally, if there exist at least $pm_{0 < p < 1}$ matched points, we say the model occurs in the scene.

4.1 Correspondence threshold computing

A Suppose in a disk of radius R, n points be placed with uniform distribution. Percus et al. [10] proved that the expected shortest distance to the nearest neighbor of a point in the disc is $R/(2\sqrt{n})$. Therefore, we choose the value of correspondence threshold as following equation:

$$cth = kR / (2\sqrt{n}) \quad \text{where} \quad 0 < k < 1 \quad (10)$$

4.2 An overview of Genetic Algorithm

The GA is a highly effective stochastic global search method that mimics the metaphor of natural biological evolution [11]. This algorithm is a computational models that generate a population of individuals (or chromosome) in each iteration, which approaches an optimal solution. Each individual is a vector in hyperspace representing one potential solution to the optimization problem. The number of elements in each hyperspace vector depend on the way to represent the problem and the number of pa-

rameters which the representation requires.

A GA is characterized by encoding system, initial population and generation procedure.

Each individual is commonly encoded to a double vector or a bit string. Usually, initialization of the population is done by seeding the population with random values and the new population is achieved using the selection, crossover and mutation procedures:

Selection The selection procedure chooses chromosomes for the next generation based on their scaled values from the fitness function. This procedure don't create new chromosome.

Crossover The crossover procedure can combine two different chromosomes to form two new individuals.

Mutation The mutation procedure create a new individual by applying the random change in a single chromosome. This procedure adds to the diversity in the evolutionary process.

5 Algorithm implementation

Two point pattern matrices M and S are the inputs of our algorithm.

Step1. Remove the mean values of two matrices and compute the radius of the two circles C_m and C_s . Set the local search area to the around of C_s . Compute the correspondence threshold using equation (10).

Step2. Determine a triple points on C_m that construct an equilateral triangle as the Referenced points .

Step3. GA-Based Subroutine :

Step3.1 Set population size to N. Although increasing generations often improves the final result, set maximum generation to G as a stopping criteria.

Step3.2 Create the initial population with uniform distribution within the local search area. Therefore, our initial population is not completely randomly, similar to [9].

Step3.3 Each triplet is encoded to double vector and used in the fitness function.

Step3.4. Apply current population to polar decomposition-based fitness function and reproduce new population using selection, crossover and mutation procedures. Here we use Roulette-wheel selection procedure and intermediate crossover procedure. For mutation process, we use gaussian function that adds a random number with gaussian distribution to each entry of single vector.

Step3.5. If number of generation is smaller than G, go to step3.3. Otherwise select the best fitness chromosome and compute it's transformation T and compensate it.

Step4. If there exist at least $p.m$ matched points, we say Model occurs in the Scene. Otherwise, mismatching occurs.

p is determined at application phase.

6 Experimental Results

We tested the algorithm on randomly generated point sets and real images. We measure the error of the algorithm by the following equation:

$$Error = norm(T_{applied} - T_{result})_2 \quad (11)$$

where $T_{applied}$ is the transformation which applied to original set to generate the secondary set and T_{result} is the transformation, returned by the algorithm.

6.1 Randomly generated point set

In the first experiment, some randomly generated point sets were matched. We used the sets which been proposed in [8], in this experiment. Table 1. shows this experiment's results.

Table 1. Performance Comparison

Algorithm Reference	[1]	[8]	This Paper
Technique	Schulz Numerical iteration	GA using Hausdorff distance	GA using matrix polar decomposition
Error/First set	1.831	0.035	0.010
Error/Second set	1.470	0.033	0.008

6.2 Fingerprint minutiae matching

Here, we applied our algorithm to a fingerprint feature matching. To solve this problem, it suffices to focus on two type of its minutiae, namely ridge endings and bifurcations [12]. We used two fingerprint images from [13] and extracted their minutiae (see Figure-2).

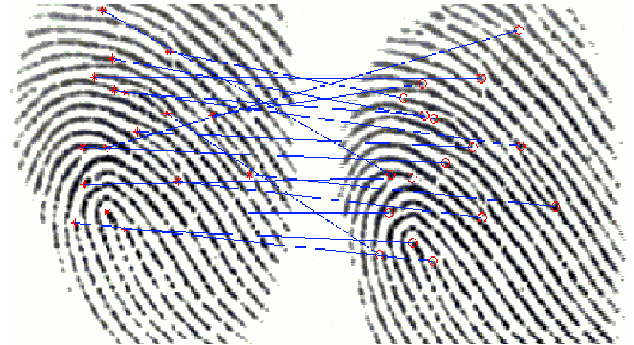


Figure 2. Initial and false correspondence between two patterns

Figure-3 gives the results on this patterns. In this figure, the matched points are very close together.

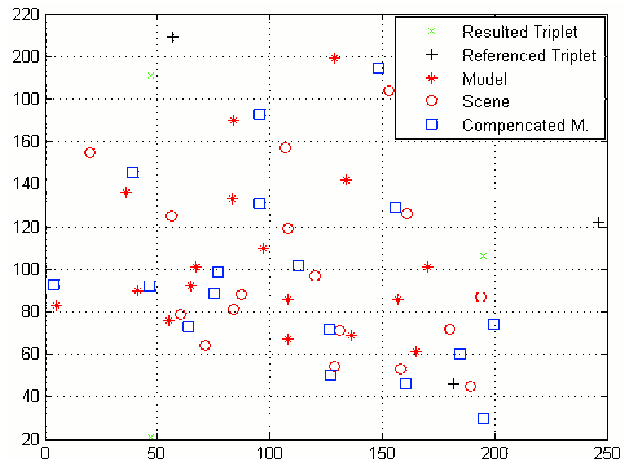


Figure 3. The Patterns and Compensated Model After 5 Generation

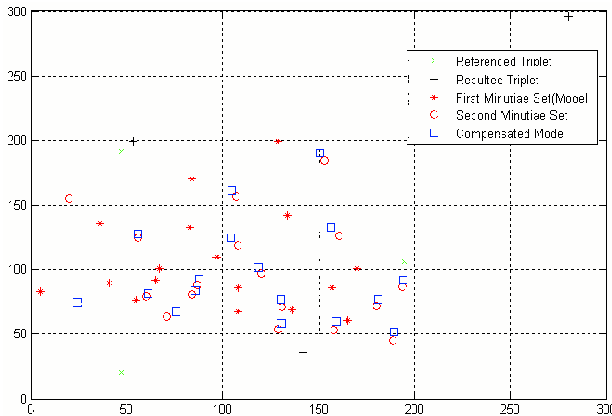


Figure 4. The Patterns and Compensated Model After 15 Generation

Therefore, we use the simple nearest neighbor rule to obtain the matched pairs.

According to the Federal Bureau of Investigation [12], to identify a fingerprint, 8 matched minutiae are enough. Since a partial fingerprint contain fewer than 20 minutiae, we select p in this application equal with $8/20$.

7 Conclusion

In this work, an algorithm is presented to solve the incomplete affine PPM problem. This algorithm uses genetic algorithm to produce an estimation of transformation matrix and uses the robust matching measure based on the matrix polar decomposition and correspondence threshold to evaluate the estimation. Because of using stochastic global search, this algorithm don't converge into local minima. We presented the algorithm in matrix form, therefore it can be applied to n-Dimensional PPM problems and can be implemented easily.

In fitness function, we correspond a desired point of compensated model to the nearest point of scene and these points are deleted before the next point correspondence. Starting from a different points may result a different correspondence, so the number of generation is sensitive to the original starting point. Therefore, to reduce the effect of this dependence on the starting point, our future works focus on definition of fuzzy correspondence which may lead to better results.

Acknowledgment This research was supported by the Iranian Telecommunication Research Centre (ITRC).

8 References

- [1] H. Li and R. Hartley, "A new and compact algorithm for simultaneously matching and estimation", IEEE ICASSP, vol. III, 2004, pp. 5-8.
- [2] H. Li, R. hartley, "Correspondence from polar decomposition", Internal Technical Report, ASST-NICTA, RSISE-ANU. Oct.2003.
- [3] M. Carcassoni , E. R. Hancock, "Spectral correspondence for point pattern matching", Pattern Recognition, vol. 36, 2003, pp.193-204.
- [4] A.Goshtasby, G.C. Stockman, "Point pattern matching using convex hull edges", IEEE Trans. Syst. Man Cybernet. SMC 15(5), 1985, 631-637
- [5] D. H. Ballard, "Generalized hough transform to detect arbitrary patterns", IEEE Trans. Pattern Anal. Mach. Intell. 13(2), , 1985, pp. 111-122
- [6] G.Stockman, S. Kopstein and S. Benett, "Matching images to models for registration and object detection via clustering", IEEE Trans. on Pattern Analysis and Machine Intelligence. PAMI 4(3), 1992, pp. 229-241
- [7] S.Gold, A. Rangarajan, C. P. Lu, S. Pappu and E. Mjolsness, "New algorithm for 2D and 3D point matching: pose estimation and correspondence", Pattern Recognition. Vol. 31, No. 8, 1998, 1019-1031
- [8] L. Zhang, W. Xu and C. Chang, "Genetic algorithm for affine point pattern matching". Pattern Recognition Letters. Vol. 24, 2003, pp. 9-19.
- [9] P.B. Van Wamelen, Z. Li and S.S. Iyengar , "A fast expected time algorithm for the 2-D point pattern matching problem", Pattern Recognition. Vol. 37, 2004, pp. 1699-1711.
- [10] A.G. Percus, O.C. Martin, "Scaling universalities of k-th nearest neighbor distance on closed manifolds", Adv. Appl. Math. Vol. 2, 1998, 424-436
- [11] M. Srinivas, Lalit M. Patnik, "Genetic Algorithms: A Survey", IEEE, 1994.
- [12] Federal Bureau of Investigation, "The Science of Fingerprint: Classification and Uses, U. S. Government Printing Office, Washington D.C., 1984
- [13] D. Maio, D. Maltoni, R. Cappeli, J.L. Wayman and A.K. Jain, " FVC2000: Fingerprint verification competition", Biolab internal report, University of Bologna, Italy, September 2000, available from <http://bias.csr.unibo.it/fvc2000/>.