



## Decision Support

## Ant colony optimization based binary search for efficient point pattern matching in images

N. K. Sreeja<sup>a,\*</sup>, A. Sankar<sup>b</sup><sup>a</sup> Department of Computer Applications, Sri Krishna College of Technology, Coimbatore 641042, India<sup>b</sup> Department of Computer Applications, PSG College of Technology, Coimbatore 641004, India

## ARTICLE INFO

## Article history:

Received 21 September 2013

Accepted 1 April 2015

Available online 12 April 2015

## Keywords:

Decision support systems

Image recognition

Point pattern matching

Ant Colony Optimization

Binary search

## ABSTRACT

Point Pattern Matching (PPM) is a task to pair up the points in two images of a same scene. There are many existing approaches in literature for point pattern matching. However, the drawback lies in the high complexity of the algorithms. To overcome this drawback, an Ant Colony Optimization based Binary Search Point Pattern Matching (ACOBSPPM) algorithm is proposed. According to this approach, the edges of the image are stored in the form of point patterns. To match an incoming image with the stored images, the ant agent chooses a point value in the incoming image point pattern and employs a binary search method to find a match with the point values in the stored image point pattern chosen for comparison. Once a match occurs, the ant agent finds a match for the next point value in the incoming image point pattern by searching between the matching position and maximum number of point values in the stored image point pattern. The stored image point pattern having the maximum number of matches is the image matching with the incoming image. Experimental results are shown to prove that ACOBSPPM algorithm is efficient when compared to the existing point pattern matching approaches in terms of time complexity and precision accuracy.

© 2015 Elsevier B.V. and Association of European Operational Research Societies (EURO) within the International Federation of Operational Research Societies (IFORS). All rights reserved.

## 1. Introduction

Point Pattern Matching (PPM) is an important problem in the field of computer vision and pattern recognition. Its major task is to pair up the points in two images of a same scene (Zhang, Zu, & Chnag, 2003). PPM can be classified as a complete match if there is a one to one mapping and an incomplete match otherwise. PPM is usually formulated as a *geometric pattern recognition* problem of point-sets, in which we seek to identify all correspondence(s) of a pattern set in a sampling set.

Some of the practical applications of PPM includes model-based tracking and recognition of a referenced template pattern in images, duplicate image identification, remotely sensed data with applications in civilian, agriculture, geology, oil and mineral exploration, astronomy, and military (Li, Meng, & Holstein, 2003).

Some of the algorithms existing in the literature for point pattern matching include Relaxation approach (Ranade & Rosenfeld, 1980), Fuzzy Relaxation approach (Ogawa, 1984), Alignment (Huttenlocher & Ullman, 1990), Randomized Alignment (Vinod & Ghose, 1993), 2D Cluster Approach (Irani & Raghavan, 1996), Nearest Neighbor search (Boxer, 1998), Probabilistic Sorted Nearest Neighbor approach

and Input Sensitive algorithms (Aiger & Kedem, 2010). The drawback in these algorithms is that the computation complexity is high. Chew et al. (1997) have proposed the best known algorithm for rigid transformations for point pattern matching which requires  $O(m^3 n^2 \log mn)$  time, where 'm' denotes the number of points in the model matched with the number of points 'n' in the scene. Van Wamelen, Li, and Iyengar (2004) have proposed Fast Expected Time (FET) algorithm which is a randomized algorithm for the alignment approach to model-based recognition. The running time of this algorithm is found to be  $O(n(\log m)^{3/2})$ , where 'm' is the number of points in the model matched with the number of points 'n' in the scene. Aiger and Kedem (2010) have proposed an Approximate Input Sensitive algorithm for point pattern matching. According to this algorithm, given point sets  $P$  and  $Q$  in the plane, the problem of point pattern matching is to determine whether  $P$  is similar to some portion of  $Q$ , where  $P$  may undergo transformations from a group  $G$  of allowed transformations. The running time of this algorithm is roughly  $O(n \log n + km \log n)$ , where 'm' denotes the number of points in  $P$  matched with the number of points 'n' in  $Q$ . However, the drawback of the rigid transformation algorithm, FET and Input Sensitive algorithms is that the time complexity for point pattern matching is higher.

Kang, Efron, Hebert, and Kanade (2009) have proposed a Re-Search algorithm to cope with confusing patterns in an indoor environment. The drawback of this algorithm is that the images which are less similar to the query image are ranked higher when compared to the

\* Corresponding author. Tel.: +91 9659425507.

E-mail address: [sreeja.n.krishnan@gmail.com](mailto:sreeja.n.krishnan@gmail.com), [dras@mca.psgtech.ac.in](mailto:dras@mca.psgtech.ac.in) (N.K. Sreeja).

images which are more similar to the query image. To overcome the drawbacks in the existing approaches, an Ant Colony Optimization based binary search approach of point pattern matching has been proposed.

According to this approach, the point pattern of an image is obtained from the edges of an image. Each point pattern has a group of point values. Consider that the incoming image and the stored image point patterns are denoted as  $I$  and  $S$ , respectively. Then,  $I = \{ip_1; ip_2; \dots; ip_m\}$  and  $S = \{sq_1; sq_2; \dots; sq_n\}$ , where  $ip_j$  and  $sq_k$  represent the point values in the incoming image and stored image point pattern, respectively. The problem is to find whether the point values in  $I$  are similar to some or all of the point values in  $S$ . An Ant Colony Optimization based Binary Search Point Pattern Matching (ACOBSPPM) algorithm is proposed for matching the point values in  $I$  with the point values in  $S$ .

In ACOBSPPM algorithm, a group of ant agents equivalent to the number of stored image point patterns is chosen where each ant agent compares the point values in the incoming image point pattern with the point values of the stored image point pattern. The ant agents have a global value which is initially zero. Each ant agent deposits pheromone to match each point value of an incoming image point pattern with the point values of the compared stored image point pattern using a binary search method. The ant agent counts the number of point values in the incoming image point pattern matching with the compared stored image point pattern. This denotes the matching point value count. A threshold matching point value count is obtained by dividing the matching point value count by the total number of point values in the incoming image point pattern. If the threshold is greater than 5 percent, then the matching point value count of the ant agent is compared with the global value. If the matching point value count is greater than the global value, the ant agent updates the global value with the matching point value count and the corresponding stored image point pattern. The process is stopped if one of the ant agents has a threshold of 95 percent or above. If none of the ant agents obtains a threshold percent of 95, the process is stopped when all the ant agents complete the matching process. Thus the stored image point pattern corresponding to the highest matching point value count in the global value is the solution.

This paper is organized as follows. Section 2 describes the related work. Section 3 describes Ant Colony Optimization. Section 4 describes the point pattern generation. Section 5 describes Ant Colony Optimization based Binary Search Point Pattern matching for image matching. Section 6 describes the mathematical model of the system. Section 7 describes a case study. Section 8 explains the computational complexity. Section 9 discusses the experimental results. Section 10 compares ACOBSPPM algorithm with the existing point pattern matching algorithms. The conclusions are presented in Section 11.

## 2. Related work

Wayman, Jain, Maltoni, and Maio (2005) have addressed the minutiae matching problem as a point pattern matching problem and proposed an operation research solution by formulating the problem as a maximization problem. Stiglmayr and Klamroth (2009) had proposed a Branch and Bound Algorithm for Medical Image Registration and formulated the problem as a Quadratic Assignment Problem. Chui and Rangarajan (2003) had proposed a Robust Point Pattern Matching algorithm for Medical Image Registration and have formulated the point matching problem as a minimization problem.

Chew et al. (1997) have proposed the best known algorithm for rigid transformations for point pattern matching which requires  $O(m^3 n^2 \log mn)$  time, where ' $m$ ' denotes the number of points in the model matched with the number of points ' $n$ ' in the scene. Aiger and Kedem (2007) have proposed a Geometric pattern matching for point

sets in a plane. The runtime of this algorithm is  $O(n^2 \log^4 mn)$ , where  $m$  and  $n$  are the number of points in  $P$  and  $Q$ , respectively.

Van Wamelen et al. (2004) have proposed Fast Expected Time (FET) which is a randomized algorithm for the alignment approach to model-based recognition. The runtime of this algorithm is found to be  $O(n(\log m)^{3/2})$  where ' $m$ ' is the number of points in the model matched with the number of points ' $n$ ' in the scene. Aiger and Kedem (2010) have proposed an Approximate Input Sensitive (AIS) algorithm for point pattern matching. The runtime of this algorithm is  $O(n \log n + km \log n)$ , where ' $m$ ' denotes the number of points in the model matched with the number of points ' $n$ ' in the scene. However, the runtimes of rigid transformation algorithm, geometric pattern matching algorithm, FET and AIS algorithms were high.

Qin, Li, and Tian (2005) have proposed a modified greedy algorithm for Remote Imagery matching by epipolar constraint and local reliability constraint for Remote Sensing Image. However, the drawback is that the precision of the final estimation of the epipolar geometric depends tightly on those of the 2D matched points. To have a better estimation of the epipolar geometric, the accuracy of the matched points should be increased and there must be good distribution of the 2D matched points.

Kang et al. (2009) proposed a Re-Search algorithm for image matching, that is designed to cope with self-repetitive structures and confusing patterns in the indoor environment. However, the drawback of this algorithm is that the images with a lower similarity to the query image were ranked higher than the images with a higher similarity to the query image.

Similarity of images can also be found using Pearson Correlation Coefficient, Spearman's rank correlation and Kendall's method. However, the computational complexity for images of size ' $n$ ' pixels for Pearson Correlation coefficient is of the order ' $n$ '. To locate a template in an image having  $N$  subimages, the time required is proportional to ' $Nn$ '. Also, the computational time becomes considerable for large values of ' $N$ ' and ' $n$ '. The computational complexity of Kendall's method is of the order ' $n^2$ ' and Spearman Rank Correlation Coefficient requires an order of  $n \log_2 n$  for images of size ' $n$ ' pixels (Goshtasby, 2012).

To overcome these drawbacks, an Ant Colony Optimization based Binary Search Point Pattern Matching (ACOBSPPM) algorithm is proposed to match an incoming image with the stored images. The advantage is that, the runtime of ACOBSPPM algorithm is less when compared to rigid transformation algorithm for point pattern matching, geometric point pattern matching algorithm, FET and AIS algorithms. Also, it is shown that the precision accuracy of ACOBSPPM algorithm is higher when compared to Re-Search algorithm. Also, the computational complexity of ACOBSPPM algorithm is less when compared to the computational complexity of correlation coefficient methods.

## 3. Ant colony optimization

A colony of ants denoting a set of computational concurrent and asynchronous agents moves through states of the problem corresponding to partial solutions of the problem to solve (Dorigo & Stutzle, 2005). They move by applying a stochastic local decision policy based on two parameters, called *trails* and *attractiveness*. By moving, each ant incrementally constructs a solution to the problem. When an ant completes a solution, or during the construction phase of the solution, the ant evaluates the solution and modifies the trail value on the components used in its solution. This pheromone information will direct the search of the future ants. Furthermore, an ACO algorithm includes two more mechanisms such as *trail evaporation* and, optionally, *daemon actions*. Trail evaporation decreases all trail values over time, in order to avoid unlimited accumulation of trails over some component. Daemon actions can be used to implement centralized actions which cannot be performed by single ant, such as the invocation of a local optimization procedure, or the update of global information to be used to decide whether to bias the search

Download English Version:

<https://daneshyari.com/en/article/6896514>

Download Persian Version:

<https://daneshyari.com/article/6896514>

[Daneshyari.com](https://daneshyari.com)