Scientific Journal of Impact Factor (SJIF): 4.72
e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406
International Journal of Advance Engineering and Research Development
Volume 4, Issue 3, March -2017

CONTOUR CORRESPONDENCE USING ANT COLONY OPTIMIZATION

Mr. B. Chandrashaker Reddy [1], V. Ushaswini [2], N. Karthik [3], R. Manasa [4]

[1] *Assistant Professor, Electronics and Communication Engineering, NNRG, Telangana, India*
[2] *Student, Electronics and Communication Engineering, NNRG, Telangana, India*
[3] *Student, Electronics and Communication Engineering, NNRG, Telangana, India*
[4] *Student, Electronics and Communication Engineering, NNRG, Telangana, India*

Abstract — *This Paper formulates contour correspondence as a Quadratic Assignment Problem (QAP), this can be solved by incorporating proximity information and order preservation. By this we can only get an appropriate solution, Hence we are going for Hungarian algorithm. Hungarian Algorithm is used to solve Assignment Problem. There are many experiments for solving this problem but it demonstrates that this approach yields high-quality correspondence results and is computationally efficient when compared to other methods.*

Keywords- *Quadratic assignment problem, Proximity, Order preservation, Hungarian Algorithm, Swarm intelligence, Stigmergy.*

I. INTRODUCTION

The action of making the best or most effective use of a situation or resource and an act, process, or methodology of making something (decision) as fully perfect, or effective as possible is known as optimization.

The first Optimization Technique is classical optimization. In this technique there are some drawbacks. Hence we go for Heuristic Optimization Technique. Heuristic optimization Technique is further classified into three types. They are:-

1. Evolution based
2. Swarm based
3. Ecology based

In geometric processing finding a meaning full mapping between shapes is fundamental problem, with many applications in computer graphics, vision, and medical imaging. In this paper, we focus on 2D contour correspondence, a classical problem in computer vision for object tracking, recognition, and retrieval, among other tasks. In medical computing, establishing point correspondence allows for statistical shape modeling and analysis of anatomical structures. Contour matching is also the first step towards planar shape morphing, which finds applications in animation and shape analysis. Even in 3D shape modeling, the matching of contours is often an integral sub problem. Also, reducing the 3D object matching problem to the matching of a set of projected object outlines was shown to be effective for 3D shape retrieval.

Optimization is divided into two classifications depending on whether variables are continuous or discrete. An optimization problem with discrete variables is known as combinatorial optimization problem. Combinatorial optimization is about defining an optimal object from finite set of objects. It operates on the domain of those optimization problems, in which the set of feasible solutions is discrete or can be reduced to discrete, and in which the goal is to find the best solution. The optimization problems with continuous variables include constrained problems.

II. METHODOLOGY

Ant Colony Optimization (ACO) is a system based agents which simulate the natural behavior of ants including mechanisms of cooperation and adaption. ACO studies artificial systems which takes the inspiration from the behavior of real ant colonies and which are used to solve discrete optimization problem.

Although the ACO metaheuristic does not guarantee convergence to a global optimum, it has been experimentally shown that ACO is one of the most successful approaches for solving structured real-life instances of the QAP. Moreover, when incorporating proximity information, solving the correspondence problem can be viewed as solving a QAP, as we have shown in the last section. In this section, we describe a novel extension of the ACO framework, which has been used for solving assignment problems, to deal with the specific shape correspondence problem.

III. ANT COLONY OPTIMIZATION

ACO was developed by Marco Dorigo in 1990. The technique to calculate shortest path between source and destination. Mimics the behaviour of Natural ants. It helps in finding out the optimized path between the source and destination. The first ant wanders randomly until it finds the food source, then it returns to the nest, laying a pheromone trails. The ants on the shortest path lay pheromone trails faster, making it more appealing to future ants. The ants become increasingly likely to follow this shortest path. The pheromone trails of the longer paths evaporate.

3.1. Ants in search of food

The sequence in which ants flow and this is used in various applications for optimization.

1) Ants go through the food while laying down pheromone trails.
2) Shortest path is discovered via pheromone trails
    - Each ant moves at random first
    - Pheromone is deposited on path
    - Shorter path, more pheromone trails
    - Ants follow the intense Pheromone trails.

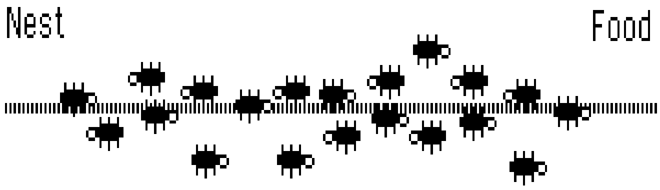

Figure 1(a)

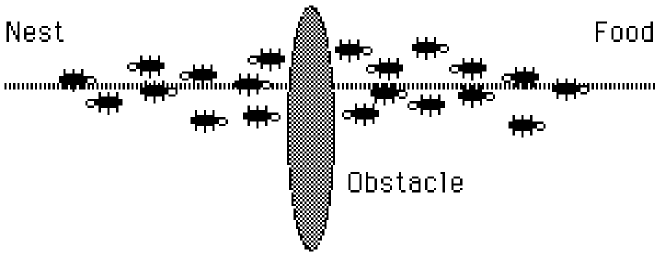

Figure 1(b)

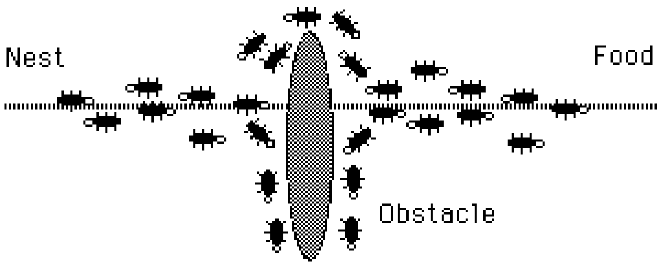

Figure 1(c)

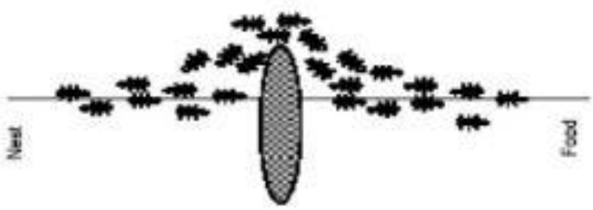

Figure 1(d)

3.1.1. Process behind ACO

Actually, ants are almost blind. They are in capable of doing there complex tasks alone. They just depend on swarm intelligence for survival. Capable of establishing the shortest route paths from their colony to feeding sources and back.

They use stigmergic communication via Pheromone trials. Follow existing pheromone trials with high probability. What emerges is a form of auto catalytic behavior the more ants follow a trail, the more attractive that trail becomes for being followed. The process is thus characterized by a positive feedback loop, where the probability of discrete path choice increases with the number of times the same path was chosen before.

- Swarm intelligence is nothing but the Collective system capable of accomplishing difficult tasks in dynamic and varied environments without any external guidance or control and with no central coordination. Achieving a collective performance which could not normally be achieved by an individual acting alone. Constituting a natural model particularly suited to distributed problem solving.
- Two individuals interact indirectly when one of them modifies the environment and the other responds to the new environment at a later time. This is stigmergy.
- Autocatalysis is a positive feedback loop that drives the ants to explore promising aspects of the search space over less promising areas.
- Pheromone trail is one of the main chemical signaling methods in which many social insects depend on, trail pheromone deposition can be considered one of the main facts to explain the success of social insect communication today. Many species of ants, including those in the genus crematograster use trail pheromones.

3.2 Resemblances and alteration with real ants

Most of the ideas of ACO stem from real ants. In particular, the use of:
1) A colony of cooperating individuals,
2) An (artificial) pheromone trail for local stigmergic communication.
3) A sequence of local moves to find shortest paths, and
4) A stochastic decision policy using local information and no look ahead.

3.3 Flowchart:

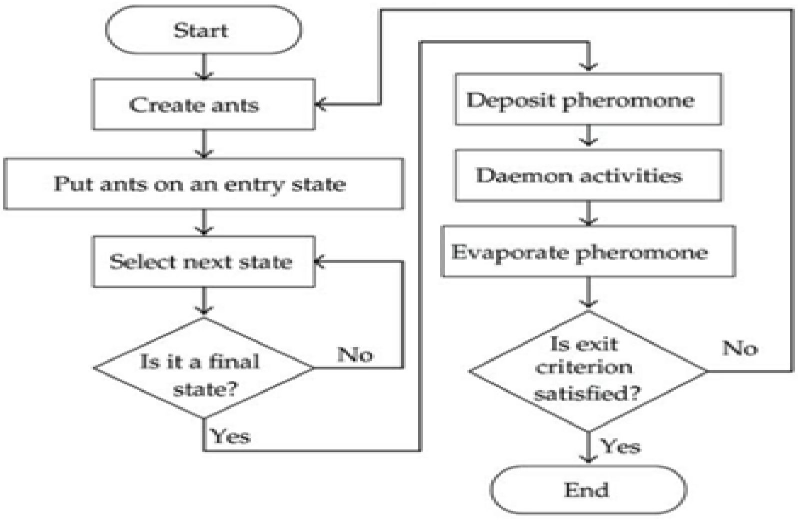

Figure 2: flowchart

IV. CONTOUR CORRESPONDENCE

We convey contour correspondence as a Quadratic Assignment Problem (QAP), incorporating proximity information. By maintaining the neighbourhood relation between points this way, we show that better matching results are obtained in practice. We propose the first Ant Colony Optimization (ACO) algorithm specifically aimed at solving the QAP-based shape correspondence problem. Our ACO framework is springy in the sense that it can handle general point correspondence, but also allows extensions, such as order preservation, for the more specialized contour matching problem. Various experiments are presented which demonstrate that this approach yields high quality correspondence results and is computationally efficient when compared to other methods.

Finding a meaningful matching between shapes is a fundamental problem in geometry processing, with many applications in computer graphics, vision, and medical imaging. In this paper, we emphasis on 2D contour correspondence, a classical problem in computer vision for object tracking, recognition, and retrieval, among other tasks. In medical computing, establishing point correspondence allows for statistical shape modeling and analysis of anatomical structures. Contour matching is also the first step towards planar shape morphing, which finds applications in animation and shape analysis. Even in 3D shape modeling, the matching of contours is often an integral subproblem. For example, surface reconstruction from CT or MRI data, or from data collected via the lofting technique, requires correspondence between contours from adjacent slices. Also, reducing the 3D object matching problem to the matching of a set of projected object outlines was shown to be effective for 3D shape retrieval.

One of the main methods to contour correspondence is to compute a shape descriptor for each selected feature point to be matched. A matching can then be extracted from the descriptors in a variety of ways, e.g., via the simple greedy best matching, solved as a bipartite matching problem using the Hungarian method, relying on variants of the iterative closest point (ICP) scheme, which is based on descriptor distances and shape alignment via rigid or non-rigid deformations, or computed by dynamic programming under point ordering.

However, a problem of all these optimization schemes, along with most other contour correspondence algorithms proposed so far, is that they treat the shape descriptors independently and do not consider proximity information measured between feature points on the same shape. For example, such information may be used to ensure that a feature pair which is close-by on one shape gets mapped to points that are also close on the other shape. Compared to methods based purely on shape descriptors, the use of proximity information, e.g., by incorporating a regularization term in the cost function, can provide a better handling of shapes with missing parts or a lack of salient features. Incorporating proximity into an optimization framework, we can formulate point correspondence via the Quadratic Assignment Problem (QAP). It is fit that QAP is one of the most difficult optimization problems to solve, yet a simple heuristic which mimics the behavior of ants has led to a great deal of success. In this paper, we adopt ant colony optimization (ACO), to solve the contour correspondence problem and make the following contributions.

4.1 ACO for contour correspondence

This section formulates the post problem and describes how the ACO metaheuristic is applied.

4.1.1 Problem formulation using QAP

Given two point sets I and J, the shape correspondence problem can be stated as finding a meaningful mapping from points of I to points of J which minimizes a given objective function. That is, we seek π^* such that

$$\pi^* = \text{argmin}_\pi(OBJ(\pi,I,J)), \qquad (1)$$

Where OBJ is the objective or cost function which evaluates the matching π in relation to the shapes characterized by I and J, and π is a mapping such that $\forall i \in I, \exists\ j \in J : \pi(i) = j$. We also assume without loss of generality that $|I| \leq |J|$.

4.1.2 The assignment problem (AP)

A common approach to shape correspondence is to extract a set of features for each point, referred to as the shape descriptors. Examples of shape descriptors include shape contexts, among many others. Next, a distance measure between two shape descriptors has to be defined. The assignment problem or AP seeks a correspondence which minimizes the sum of the distances between descriptors of points on one set and the descriptors of the corresponding points on the other set:

$$AP(\pi,R,I,J) = \sum_{i \in I} D_R(Ri, R_{\pi(i)}) \qquad (2)$$

Where Ri denotes the descriptor at point i, and DR is the shape descriptor distance measure. Here, the matching is constrained to be one-to-one (not necessarily onto). The optimal matching for this cost measure can be computed in cubic time with the Hungarian algorithm.

4.1.3 Incorporating proximity information

Another element for evaluating a correspondence between continuous shapes, e.g., contours or surfaces, is the preservation of quantitative neighborhood or proximity information. Namely, if point i from shape I and point j from shape J are matched, then a close-by neighbour i^* of i on I should be matched with a point j^* on J that is close to j. Berg et al. refer to this as the minimization of distortions in a correspondence. This differs from the case of order preservation in that besides point ordering, we are also concerned with distances between point pairs on the same shape.

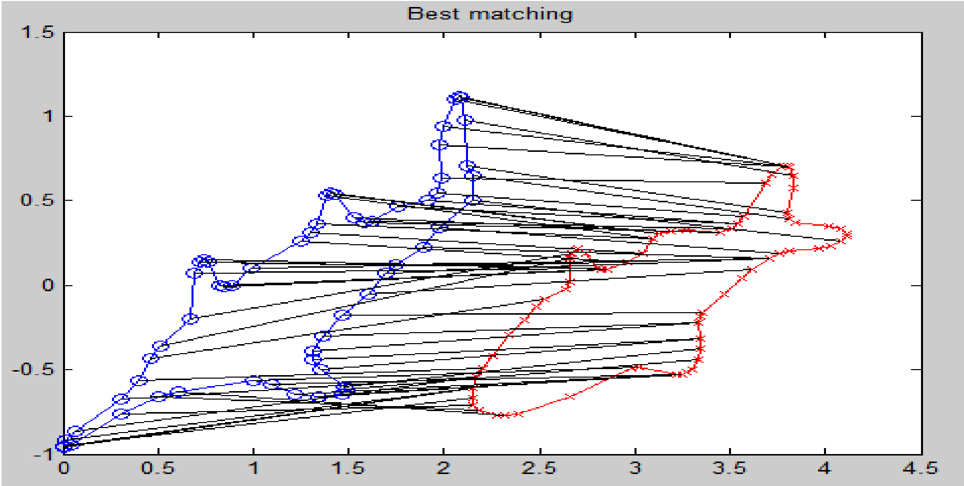

Figure 3: Proximity

4.1.4 Order preservation for contour matching:

Note that shape descriptors might not be the only factor one should take into account when evaluating a correspondence. One possible element pertaining to contour matching is order preservation, which follows from the observation that the vertices defining a contour are ordered. For example, the COPAP algorithm solves the cyclic-orderpreserving assignment problem via dynamic programming. It has been shown that order-preserving contour matching significantly improves the correspondence results. However, it is unclear how it can be extended to other domains, e.g., for points residing on a 3D shape, for which a canonical point ordering is not available.

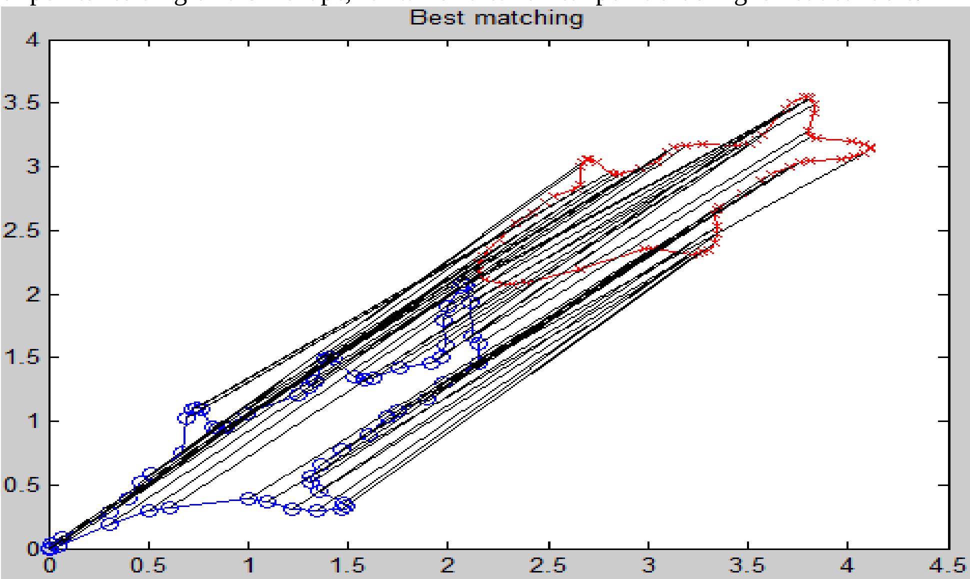

Figure 4: orderpreservation

4.2 Correspondence algorithm using ACO

Although the ACO metaheuristic does not guarantee convergence to a global optimum, it has been experimentally shown that ACO is one of the most successful approaches for solving structured real-life instances of the QAP. Moreover, when incorporating proximity information, solving the correspondence problem can be viewed as solving a QAP, as we have shown in the last section. In this section, we describe a novel extension of the ACO framework, which has been used for solving assignment problems, to deal with the specific shape correspondence problem.

4.2.1 Graph model and correspondence extraction

We define the graph G = {V,E} that is to be traversed by the ants as a complete, directed bipartite graph. The set of vertices of this graph is composed of the two point sets to be matched, i.e., V = I ∪J. The directed edges E fully connect the two point sets. The path that is determined by the traversal of an ant on G corresponds to a possible solution to the assignment (correspondence) problem. During such a traversal, two conceptually distinct tasks are performed. When an ant traverses a directed edge that connects a vertex in I to a vertex in J, an assignment from I to J is determined. On the other hand, when an ant traverses an edge from J to I, the order in which the vertices are assigned is determined. An ant starts the graph traversal from a randomly selected vertex. It traverses edges until an assignment of each vertex in

I is determined (recall that $|I| \leq |J|$). The final assignment π is given by the edges from I to J chosen by the ant. Note that a correspondence obtained in this way is not necessarily one-to-one, as opposed to the AP or COPAP matching formulations.

4.2.2 Path construction and evaluation

When traversing from a vertex $i \in I$ to a vertex in J, the probability p^k_{ij} of an ant k choosing the edge that connects to vertex $j \in J$ is given by edge probability is,

$$P^k_{ij} = \frac{\alpha\tau_{ij} + (1-\alpha)\eta_{ij}}{\sum_{l \in Ni} [\alpha\tau_{il} + (1-\alpha)\eta_{il}]} \quad (3)$$

where τ_{ij} quantifies the pheromones accumulated on edge (i, j), η_{ij} indicates the desirability (or probability) of traversing (i, j) based on heuristic information, and $Ni = \{l \in J : (i,l) \in E\}$ is the immediate neighborhood of vertex i. The parameter $0 \leq \alpha \leq 1$ regulates the influence of pheromones over heuristic information. We can see that the choice of the traversed edge is stochastic, where the sum of probabilities $\sum_{j \in Ni} p^k_{ij} = 1$. Moreover, when traversing back from J to I, an edge pointing to any vertex in I that has not been visited yet can be chosen. Each edge from j to I has the same probability of being selected, and no pheromones or heuristic information are considered. This choice is certainly not unique and our framework is quite flexible in allowing for other variants.

4.2.3 Pheromone updates

Pheromones are updated at the end of an ACO iteration. First, pheromones are evaporated at a constant pheromone evaporation rate ρ, $0 \leq \rho \leq 1$,

4.2.4 Pheromone evaporation:

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} \quad (4)$$

for all edges (i, j). Next, new pheromone is deposited only on edges that were traversed by the ants,

4.2.5 Pheromone deposition:

$$\tau_{ij} \longrightarrow \tau_{ij} + \sum_{k=1}^{M} \Delta\tau^k_{ij} \quad (5)$$

where $\Delta\tau^k_{ij}$ is the amount of pheromone that ant k has deposited on edge (i, j). It is given as a constant δ (a free parameter) divided by the correspondence cost defined above: the larger the cost, the less the amount of deposited pheromones. In addition, a minimum level of pheromones τmin is maintained on all edges to avoid completely eliminating certain traversals during the ants' exploration.

The main function for our ACO based shape correspondence algorithm

```
initGraph(I,J)
    G ← A complete, directed, bipartite graph between I and J
    for each edge e in {I → J} ⊂ E(G) do
        e_pher ← τ0
    end for
    for each edge e in {J → I} ⊂ E(G) do
        e_pher ← 0
    end for
    return G
```

Matching construction function

```
constructMatching(G)
    M ← ∅
    i ← Randomly chosen from the vertices in I
    while there are unmatched vertices in I do
        for each vertex j in J do
            /* for order preservation */
            if j is in the set of valid J vertices then
                P[j] ← (α τ_ij + (1−α) η_ij) / (Σ_{l∈N} α τ_il + (1−α) η_il)
            else
                P[j] ← 0
            end if
        end for
        j_choice ← Probabilistically chosen vertex in J
                   according to probabilities P
        M ← M ∪ {i, j_choice}
        i ← Randomly chosen from the set of available
            vertices in I
    end while
    return M
```

Pheromone update

```
updatePheromones(G, Matchings)
    /* Evaporate pheromone */
    for each edge e in {I → J} ⊂ E(G) do
        e_pher ← (1 − ρ) e_pher
    end for
    /* Add new pheromone */
    for each {M,C} in Matchings do
        Δ_pher ← δ/C
        for each edge e in {I → J} in matching M do
            e_pher ← e_pher + Δ_pher
        end for
    end for
    /* To ensure minimum pheromone level */
    for each edge e in {I → J} ⊂ E(G) do
        if e_pher < τ_min then
            e_pher ← τ_min
        end if
    end for
```

Here we get approximate output to get optimal solutions we are going for Hungarian algorithm. The steps involved in Hungarian algorithm are given.

V. HUNGARIAN ALGORITHM

The Hungarian method is a combinatorial algorithm that solves the assignment problem in polynomial time and which anticipated later primal-dual methods. It was developed and published in 1955 by Harold Kuhn, who gave the name "Hungarian method" because the algorithm was largely based on the earlier works of two Hungarian mathematicians: Dénes Kőnig and Jenő Egerváry.

For solving assignment problems Hungarian a5lgorithm is used. We have two types of assignment problem. They are 1) Balanced assignment problem and 2) Unbalanced assignment problem. This Hungarian Algorithm is used to solve only balanced assignment problem. Balanced assignment problem is nothing but number of rows should be equal to the number of columns.

5.1 Steps involved in Hungarian Algorithm:

- **Phase1:** Row and column reductions

Step1:- Subtract the minimum value of each row from the entries of thet row.
Step2:- Subtract the minimum value of each column from the entries of thet column.

• **Phase2:** Optimization of the problem

Step1:- Draw a minimum number of lines to cover all the zeroes of the matrix.

Procedure:

a) Row scanning:-

1) Starting from the first row ask the following questions. Is there exactly one zero in that row? If yes, mark a square around that zero entry and draw a vertical line passing through that zero, otherwise skip that row.
2) After scanning the last row check whether all the zeroes are covered with lines. If yes, go to step 2, otherwise, do column scanning.

b) Column scanning:-

1) Start from the first column, ask the following question. Is there exactly one zero in that column? If yes, mark a square around that zero entry and draw a horizontal line passing through that zero, otherwise skip that column.
2) After scanning the column, check whether all the zeroes are covered with lines.

Step2:- Check whether the number of square marked is equal to the number of rows of the matrix. If yes, go to step 5, otherwise, go to step 3

Step3:- Identify the minimum value of the undeleted cell values.

a) Add the minimu undeleted cell value at the intersection point of the present matrix.
b) Subtract the minimum undeleted cell value from all the undeleted cell values.
c) All other entries remain same.

Step4:- Go to step 1 and repeat the procedure until you get optimal solution.

Step5:- Optimal solution is obtained.

VI. RESULT

After incorporating proximity information and order preservation we got an approximate output. Then the errors are given to the Hungarian algorithm to get optimal solution.

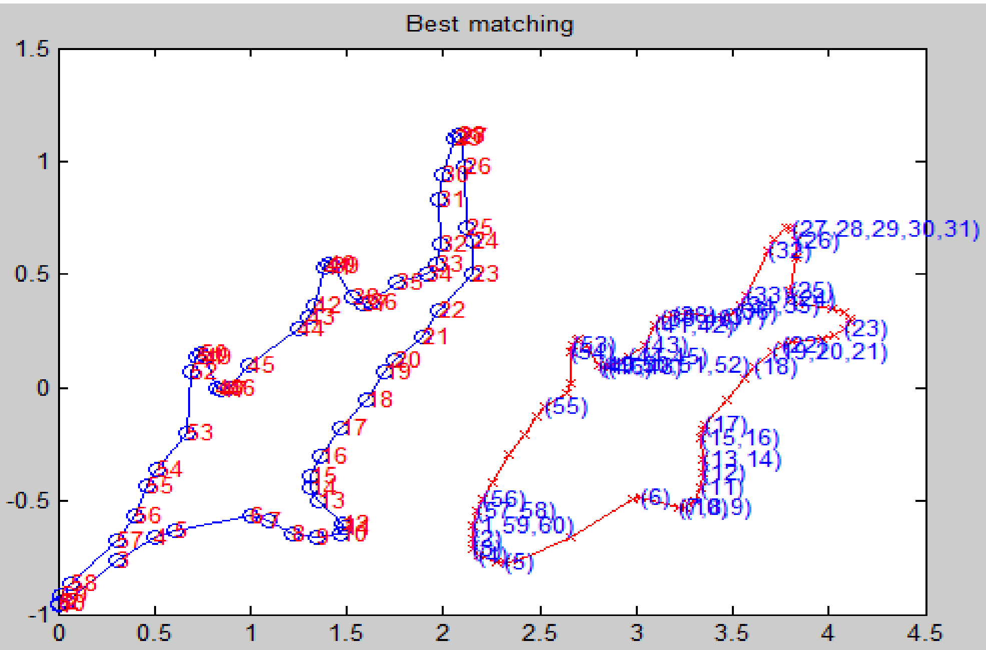

Figure 5:output

VII. CONCLUSION

We formulate shape correspondence as a Quadratic Assignment Problem (QAP), incorporating proximity information into the point matching objective function. We also propose the first Ant Colony Optimization (ACO) algorithm directly aimed at solving the point and contour correspondence problems, which are difficult problems when the QAP formulation is adopted. The advantage of incorporating proximity and the effectiveness of the proposed method have been verified with a set of experiments. Qualitative and quantitative results show that the correspondences obtained by our ACO algorithm are at least comparable to those obtained by the best alternative methods, e.g.,. In several cases, we have demonstrated clear advantages offered by our approach. In addition, our algorithm also has the advantage that its resource requirements scale moderately for contours of increasing size. Although the descriptor used in our experiments did not allow to compute matchings for planar shapes with articulated deformations, the original shape context can be extended using geodesic neighborhood information, this can be incorporated into our ACO framework easily, as well as any shape descriptor that allows to compute the similarity between vertices on two shapes.

REFERENCES

[1] A. C. Berg, T. L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In Proc. IEEE Conf. on CVPR, pages 26–33, 2005.
[2] D. Meyers, S. Skinner, and K. Sloan. Surfaces from contours. ACM Trans. on Graphics, 11(3):228–258, 1992.
[3] T. B. Sebastian, P. N. Klein, and B. B. Kimia. On aligning curves. PAMI, 25(1):116–124, 2003.
[4] A. Tal and G. Elber. Image morphing with feature preserving texture. Computer Graphics Forum, 18:339–348, 1999.

