

# Edge detection in images using Ant Colony Optimization Algorithms

---

Arjun Varshney  
November 16, 2015

## 1 THINGS DONE SO FAR

In this project, we use Ant Colony Optimization (ACO) algorithms [1] to detect edges in an image and compare results of various algorithms for a set of images. ACO aims to find the optimal solution to the target problem through a guided search (i.e., the movements of a number of ants) over the image by constructing the pheromone information (matrix of pixel size). To be more specific, suppose totally  $K$  ants are applied to find the optimal solution in a space  $\chi$  that consists of  $M_1 \times M_2$  nodes, the procedure of ACO can be summarized as follows:

- Initialize the positions of totally  $K$  ants, as well as the pheromone matrix  $\tau^{(0)}$ .
- For the construction-step index  $n = 1 : N$ ,
  - For the ant index  $k = 1 : K$ ,
    - \* Consecutively move the  $k$ -th ant for  $L$  steps, according to a probabilistic transition matrix  $\mathbf{p}^{(n)}$  (with a size of  $M_1 M_2 \times M_1 M_2$ ).
  - Update the pheromone matrix  $\tau^{(n)}$ .
- Make the solution decision according to the final pheromone matrix  $\tau^{(N)}$ .

I have implemented the Ant Colony System meta-heuristic for detecting the edges, which is different from other ACO metaheuristics, as it has a local pheromone update too in addition to the pheromone update performed at the end of the construction process (offline pheromone update)[1]. The algorithm is broken into four parts, initialization, construction, update and decision process.

## 1.1 INITIALIZATION

We start with initializing the ants to move on an image for constructing the pheromone matrix. The initialization is done randomly in the [2], but I have tried various methods (see preliminary results) to find the best position of ants so that the image formation is better and faster. The pheromone matrix has the information of the edge at each pixel location of the image. The movements of the ants are guided by the local variation of the image's intensity values. This procedure runs for N iterations to construct the pheromone matrix by iteratively performing both the construction and the update process. The decision process is made to find the edge with the updated pheromone matrix values.

## 1.2 CONSTRUCTION

At the  $n^{\text{th}}$  step, an ant is randomly selected, and this ant is made to move for M steps. This ant moves from the current node to its neighboring node according to a transition probability function. The permissible range of ant's movement is determined by two types of neighborhood (4-connectivity, or 8 connectivity) as shown below.

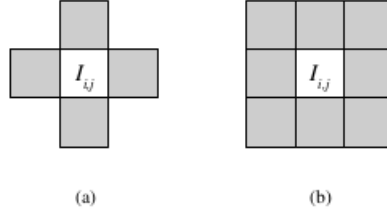


Figure 1.1: a) 4-connectivity and b) 8-connectivity neighborhood

## 1.3 UPDATE

The update process is a two step process for updating the pheromone matrix. The first update is done locally after each ant moves a step. Each element of the pheromone matrix associated to that location of the ant is updated to,

$$\tau_{i,j}^{(n-1)} = \begin{cases} (1 - \rho) \cdot \tau_{i,j}^{(n-1)} + \rho \cdot \Delta_{i,j}^{(k)} & \text{if } (i,j) \text{ is visited by the current } k^{\text{th}} \text{ ant;} \\ \tau_{i,j}^{(n-1)} & \text{otherwise} \end{cases}$$

where  $\tau_{i,j}^{(n-1)}$  is the pheromone value at the  $(i,j)$  position of the pheromone matrix,  $\rho$  is the evaporation rate and  $\Delta_{i,j}^{(k)}$  is determined by the heuristic matrix.

The second update is the global update after all the ants finish one construction step. The pheromone matrix is then globally updated by,

$$\tau^n = (1 - \psi) \cdot \tau^{(n-1)} + \psi \cdot \tau^{(0)}$$

where  $\psi$  is the pheromone decay coefficient.

## 1.4 DECISION

In this portion, a binary decision is made at each pixel to determine whether it is edge or not, by applying a threshold  $T$  on the final pheromone matrix  $\tau^{(n)}$ . The initial threshold  $\tau^{(0)}$  is chosen to be the mean value of the pheromone matrix. The entries of the pheromone matrix is labeled into two class according to the criterion that its value is lower than  $\tau^{(0)}$  or larger than  $\tau^{(0)}$ . The new threshold is calculated as the average of two mean values of each of above two categories. The above process is repeated till there is no change in the threshold value.

## 2 PRELIMINARY RESULTS

I ran the Ant Colony System algorithm on the gray scale versions of few of my pencil sketches. The results of the Ant Colony System meta-heuristic are as shown below:

- following results were obtained on placing the ants initially on the random rows and columns of the image. As expected, initial results were not so good due to bad initial placements of the ants.

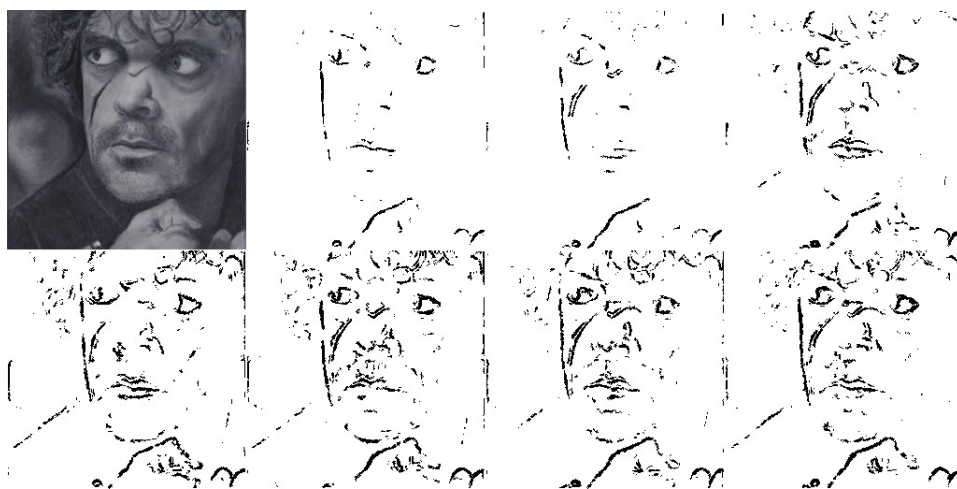


Figure 2.1: Tyrion : Ants initialized on random rows and columns of the image

- following results were obtained on placing the ants initially on the diagonal of the pheromone matrix. The initial result shows clearly that the image edges are detected starting from the diagonal and the final result was found satisfactory.

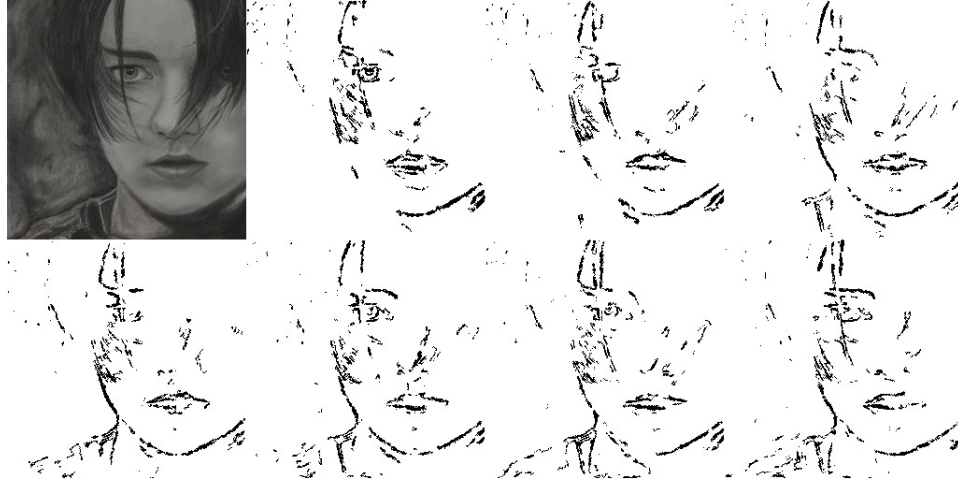


Figure 2.2: Arya : Ants initialized on the diagonal of the image

- following results were obtained on placing the ants initially on the highest values of the heuristics. The initial results shows clearly that the edges are detected in the first iteration itself. More edges can be identified easily in fewer iterations.

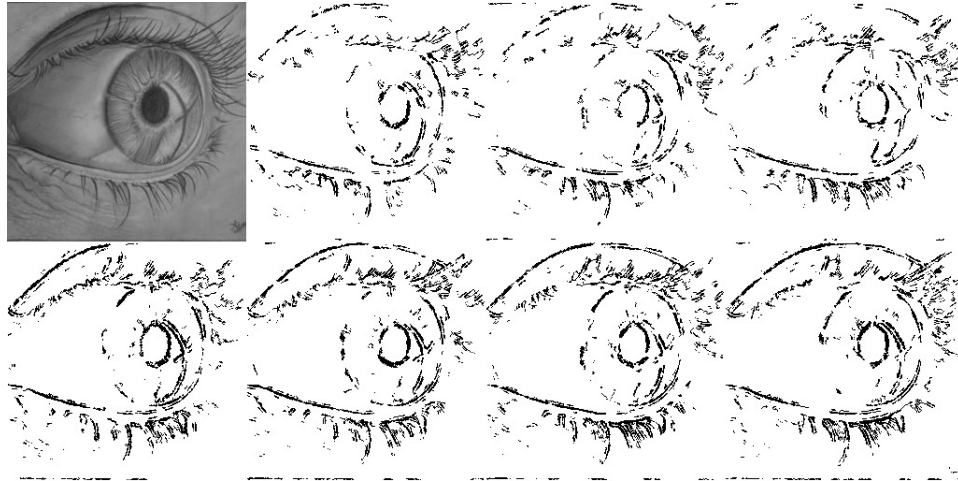


Figure 2.3: Arya : Ants initialized on the highest values of the heuristic matrix

### 3 REMAINING WORK

The plan is to improve the current Ant Colony System Implementation using the initialization of the ants or tweaking some parameters like the threshold as stated in the decision process. Also, Ant System [3] meta-heuristic is yet to be implemented. The comparison of the two algorithms with respect to execution time and the image results, would be the final step of this project.

## REFERENCES

- [1] Jing Tian, Weiyu Yu, and Shengli Xie, *An Ant Colony Optimization Algorithm For Image Edge Detection*, 2008 IEEE Congress on Evolutionary Computation (CEC 2008)
- [2] Marco Dorigo and Krzysztof Socha, *An Introduction to Ant Colony Optimization*, April 2006
- [3] H. Nezamabadi-Pour, S. Saryazdi, and E. Rashedi, *Edge detection using ant algorithms* Soft Computing, vol. 10, pp. 623-628, May 2006.