Digital Image Edge Detection using Enhanced Ant Colony Optimization Technique

Jeetu Singh
M.Tech, Department of CSE
Jaypee Institute of Information Technology University
Noida, India

Ankit Vidyarthi
Research Scholar, Department of CSE
Malaviya National Institute of Technology Jaipur
Jaipur, India

ABSTRACT

The detection of the edge is one of the important part in the field of Image Processing. In this paper we proposed an improved ACO algorithm for digital images edge classification. The classification is basically done as per the natural phenomenon of the movement of ants for searching paths. We have proposed a new modified ACO algorithm for better visual effects and compared the experimental results with previous standard one.

Keywords- Ant colony optimization, Edge Detection.

1. INTRODUCTION

Ant colony optimization (ACO) is a nature-inspired optimization algorithm [1], [2], motivated by the natural phenomenon that ants deposit pheromone on the ground in order to mark some favorable path that should be followed by other members of the colony. Many algorithms have been developed on ACO [3] like Max-Min ant system [4] and the Ant Colony System [5] to detect an edges from digital images. Since edge is an important feature in an image and carries important information about the objects present in the image. Extraction of edges is known as edge detection. Edge detection aims to localize the boundaries of objects in an image and significantly reduces the amount of data to be processed. In this paper, an Enhanced ACO-based algorithm for image edge detection is proposed. We proposed a new enhanced and modified approach which uses the concept of evaporation rate with pheromone information, also for enhancement in calculations we uses some more mathematical functions. The main focus and stress is given on the visual perspective to detect the edges more clearly.

The remainder of the paper is organized as follows. In section II, we present the basis idea of Ant Colony Optimization Technique. In section III, a proposed enhanced Ant Colony Optimization algorithm is explained. In section IV, an experimental result are shown and finally section V, concludes the paper.

2. THE SUMMARY OF ACO ALGORITHM

An Ant Colony Optimization algorithm (ACO) is essentially a system based on agents which simulate the natural behavior of ants, including mechanisms of cooperation and adaptation [6]. In many ant species, ant walking to and from a food source deposit some substance on the ground called pheromone. Different ants adopt different paths to reach the food source and deposit the pheromone based on the fact that higher concentration of pheromone is deposited on shorter paths and smaller concentration on longer paths.

2.1. Ant Colony Optimization Algorithm

The basis procedure of ACO is given as:

- Each path followed by an ant is associated with a candidate solution for a given problem.
- When an ant follows a path, the amount of pheromone deposited on that path is proportional to the quality of the corresponding candidate solution for the target problem.
- When an ant has to choose between two or more paths, the path(s) with a larger amount of pheromone have a greater probability of being chosen by the ant.

As a result, the ants eventually converge to a short path, hopefully the optimum or a near-optimum solution for the target problem.

2.2. Edge Detection using ACO

Image edge detection refers to the extraction of the edges in a digital image. The main idea of image edge detection based on ACO is to use the number of ants to move on an image 2-D plane to create a pheromone matrix, each entry of that shows the edge information in each pixel location of the image. In addition, the movements of the ants are directed through the local variation of image intensity values. It is a process whose aim is to identify points in an image where discontinuities or sharp changes in intensity occur. The purpose of detecting sharp changes in image intensity is to capture significant events and changes in the physical properties of the world.

An ACO-based approach has the potential of overcoming the limitations of conventional methods. Furthermore, it can readily be parallelized, which makes the algorithm easily adaptable for distributed systems [7]. Several ACO-based approaches to the edge detection problem have been proposed [8]-[11]. Previously reported ACO-based approaches to image edge detection, to the best of the author’s knowledge, all use a decision rule that is based on AS. AS is the first ACO algorithm. Since its development, a number of extensions have emerged. One of the successful ones is ACS.

Edges characterize boundaries and so have a wide range of useful applications such as segmentation and identification of objects in scenes, machine vision, astronomy and microscopy imaging to name a few. Edge detection is a technique for marking sharp intensity changes, and is important in further analyzing image content. It lays the foundation for image fusion, shape extraction, image segmentation, image matching, and image tracking. Many traditional edge detection approaches result in broken pieces, which lead to incomplete resultant images. The research done in this paper as well as by many fellow researchers present that ant colony optimization based mechanism tends to compensate those lost edges. The proposed procedure indicates that the edge detection approach is efficient on compensating broken edges and more efficient than the traditional ACO approach in computation reduction.

3. PROPOSED APPROACH

The proposed approach is basically the modification of the previously used ACO algorithms for edge detection [4-11]. The modification is being done in two domains of previously existing one ACO algorithm, i.e. modification in construction process and modification in decision process and rest of all the process is kept same as described in original ACO algorithm [9]. The basic flow structure of the algorithm is shown in fig.1.

3.1. Modified Construction Process

In this step modification done is in formation of new clique and addition of two new kernel functions. In the construction process, selected ant will move on the image for L movement-steps & this process will be repeated in each construction step and till each ant moves on image. The selected ant moves from node (l, m) to its neighboring node (i, j) according to the transition probability defined as follows:

$$P_{(l,m),(i,j)}^{(n)} = \frac{\tau_{(i,j)}^{(n-1)^{\alpha}} (\eta_{(i,j)})^{\beta}}{\sum_{(i,j)\in\Omega_{(l,m)}} (\tau_{(i,j)}^{(n-1)})^{\alpha} (\eta_{(i,j)})^{\alpha}}$$

Where $\tau_{i,j}^{(n-1)}$ is the pheromone value of node (i, j), $\Omega_{(l,m)}$ the neighborhood nodes of the nodes (l, m),$\eta_{i,j}$ represent the heuristic at the node (i, j). The constants α and β represents the influence of pheromone information and heuristic information respectively.

3.2. Decision Process

Here the modification is in the selection of new threshold is being done. A pheromone matrix is used to classify each pixel either as an edge or a non-edge. The decision is made by applying a threshold T on the final pheromone. The threshold T is computed based on Otsu technique which is described in [12].

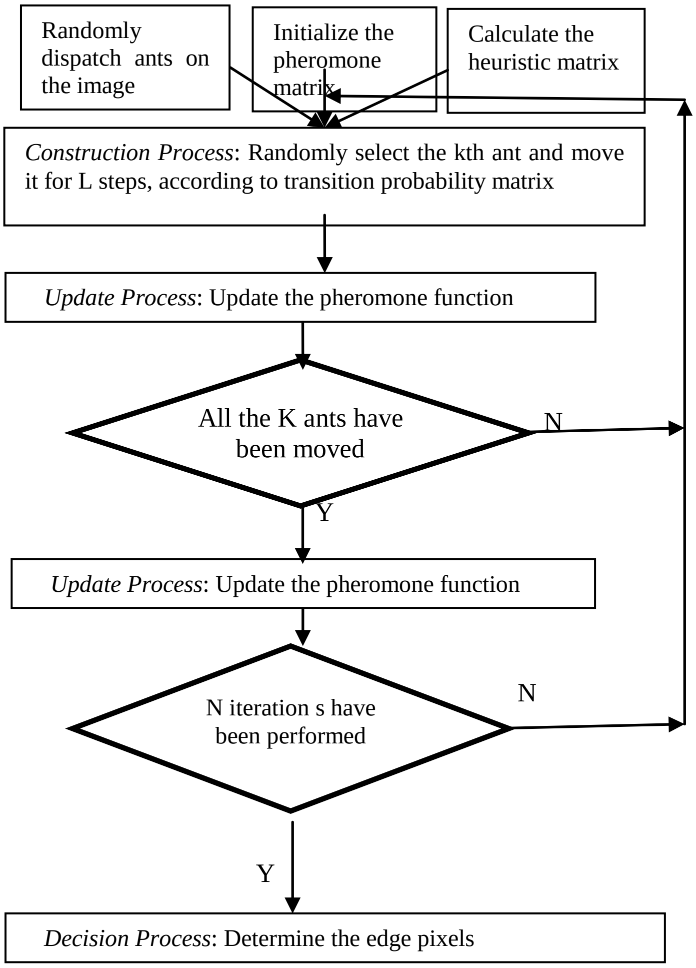
Randomly dispatch ants on the image
Initialize the pheromone matrix
Calculate the heuristic matrix
Construction Process: Randomly select the kth ant and move it for L steps, according to transition probability matrix
Update Process: Update the pheromone function
All the K ants have been moved
N
Y
Update Process: Update the pheromone function
N iteration s have been performed
N
Y
Decision Process: Determine the edge pixels

Fig.1. Flow Structure of ACO algorithm [13]

The initial threshold T (0) is selected as the mean value of the pheromone matrix. The entries of the pheromone matrix are classified in two parts as:

(a) Those entries of pheromone matrix whose value is less than T (0).

(b) Those entries of pheromone matrix whose value is larger than T (0).

Then the new threshold is calculated by taking the square of two mean values of each of the two categories and then taking their average. The complete process is repeated again and again until the threshold value becomes constant (in terms of user defined tolerance).

The whole procedure of modified ACO is given as:

Assumption: we use the 10 construction step with 40 artificial ants per construction step.

1. Pheromone matrix: This matrix contains the value of pheromone intensity which attracts the ants to follow paths traversed by other ants. Pheromone matrix are updated twice, once after the movement of each ants and secondly after movement of all the ants.

2. Probabilistic transition matrix: The value of probability for the ant's movement from one pixel to another is stored in probability transition matrix.

The procedure of ACO can be summarized as follows:

a. Let total K ants are applied to find the optimal solution in a space [χ] that consists of M1× M2 nodes.

b. Initialize the positions of total K ants, as well as the pheromone matrix τ(0).

c. For the construction-step index n = 1 : N and for the ant index k = 1: K. The selected ant moves from node (l, m) to its neighboring node (i,j) according to the transition probability that is defined as follows:

$$P_{(l,m),(i,j)}^{(n)} = \frac{\tau_{(i,j)}^{(n-1)^{\alpha}} (\eta_{(i,j)})^{\beta}}{\sum_{(i,j)\in\Omega_{(l,m)}} (\tau_{(i,j)}^{(n-1)})^{\alpha} (\eta_{(i,j)})^{\alpha}}$$

Where Ţ(i, j) is the pheromone value of node (i, j), Ω(l, m) is the neighborhood nodes of the nodes (l, m), Ƞi,j represents the heuristic information at the node (i ,j). The constants α and β represent the influence of pheromone information and heuristic information respectively.

d. Update the pheromone matrix τ(n) after the movement of each ant within each construction step.

e. To calculate the authorized amount of ant movements (i.e. Ω(l, m)) at the position(l, m), either the 4-connectivity neighborhood or the 8-connectivity neighborhood can be used.

3. Decision Process

a. Threshold is initialized as:

$$T^{(0)} = \frac{\sum_{i=1:M1} \sum_{j=1:M2} \tau_{i,j}^{(N)}}{M1M2}$$

As set the iteration index as l=0

b. Pheromone matrix τ(n) is separated in two classes where first class contains the entries of τ whose value is below T(l) and other class in which values of τ are above T(l) . Mean of the two categories (specified earlier) is calculated as:

$$m_L^{(l)} = \frac{\sum_{i=1:M1} \sum_{j=1:M2} g_{T(l)}^{L} (\tau_{i,j}^{(N)})}{\sum_{i=1:M1} \sum_{j=1:M2} h_{T(l)}^{L} (\tau_{i,j}^{(N)})}$$

$$m_U^{(l)} = \frac{\sum_{i=1:M1} \sum_{j=1:M2} g_{T(l)(\tau_{i,j}^{(N)})}^{L}}{\sum_{i=1:M1} \sum_{j=1:M2} g_{T(l)\tau_{i,j}^{(N)}}^{L}}$$

Where,

$$g_{T(l)}^{L}(x) = \begin{cases} x, & if\ x \le T^{(l)}; \\ 0, \text{otherwise}. \end{cases}$$

$$h_{T(l)}^{L}(x) = \begin{cases} 1, & if\ \ x \le T^{(l)} \\ 0, otherwise. \end{cases}$$

$$g_{T(l)}^{U}(x) = \begin{cases} x, & if\ x \ge T^{(l)} \\ 0 & otherwise \end{cases}$$

$$h_{T(l)}^{U}(x) = \begin{cases} x, & \text{if } x \ge T^{(l)} \\ 0 & \text{otherwise} \end{cases}$$

c. New threshold is updated as

$$T^{(l)} = \frac{m_L^{(l)}/m_L^{(l)}}{2}$$

d. If |T(l) − T(n−1)| > € , then go to Step 2; otherwise, the iteration process is terminated and a binary decision is made on each pixel position (i, j) to determine whether it is edge (i.e., Ei,j = 1) or not (i.e., Ei,j = 0), based on the criterion.

$$E_{i,j} = \begin{cases} 1, & \text{if} \tau_{i,j}^{N} \ge T^{(l)}; \\ 0 & \text{otherwise.} \end{cases}$$

4. EXPERIMENTAL RESULTS

The proposed ACO approach is implemented using the java programming language. Experiments are conducted to evaluate the performance of the proposed approach using four test images, i.e. human body, Building, natural scenery and house which was shown below. The modified results are being compared with the previous ACO algorithm and the computational results are being displayed.

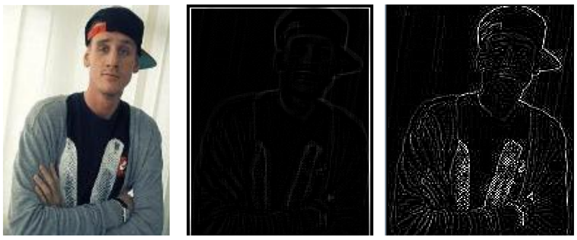

Fig. 2 (a) (b) (c)

Fig. 2 represents the first comparative result of human body.
(a) Shows the original RGB image, (b) shows the existing ACO results, (c) our approach

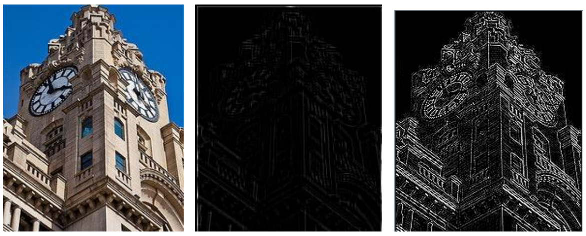

Fig. 3 represents the second comparative result of building.
(a) Shows the original RGB image, (b) shows the existing ACO results, (c) our approach

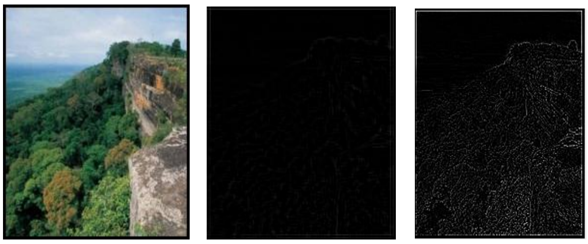

Fig. 4 represents the third comparative result of natural scenery.
(a) Shows the original RGB image, (b) shows the existing ACO results, (c) our approach

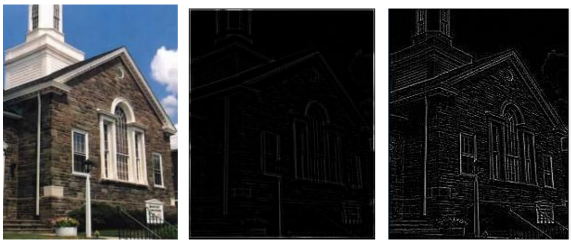

Fig. 5 represents the fourth comparative result of house.
(a) Shows the original RGB image, (b) shows the existing ACO results, (c) our approach

5. CONCLUSION

In this paper an improved ACO based image edge detection approach has been suggested. The proposed modified approach yields superior subjective performance to that of the previous existing one. Experimental results clearly show the superior performance of proposed approach with suitable parameters. The visual impact of images with respect to the edge detection is being improved. After the successful implementation of the algorithm, the edges are being sharpened and thus increase the visual effect as compared to existing basic ACO. Also our approach can be applied on any of the existing domains as shown in computational results. The continuation research should concentrate on to reduce the computational time of the proposed approach and also on more effective way to calculate heuristic information.

6. REFERENCES

[1] M. Dorigo and S. Thomas, Ant Colony Optimization. Cambridge: MIT Press, 2004.

[2] H.-B. Duan, Ant Colony Algorithms: Theory and Applications. Beijing: Science Press, 2005.

[3] S. L. Hegarat-Mascle, A. Kallel, and X. Descombes, Ant colony optimization for image regularization based on a nonstationary Markov modeling, IEEE Trans. on Image Processing, Mar. 2007, vol. 16, pp. 865–878.

[4] Y. T. Kim, Contrast Enhancement Using Brightness Preserving Bi-Histogram Equalization, IEEE Trans.,Consumer Electronics, 1997, vol. 43, no. 1, pp. 1-8.

[5] H. Ibrahim, and N. S. P. Kong, Brightness Preserving Dynamic Histogram Equalization for Image Contrast Enhancement, IEEE Trans., Consumer Electronics, Nov. 2007, vol. 53, no. 4, pp. 1752–1758.

[6] Rafael S. Parpinelli, Heitor S. Lopes, Alex A. Freitas, Data Mining with an Ant Colony Optimization Algorithm , IEEE Trans., Data Mining, 2010.

[7] Anna Veronika,C.Baterina,Carlos. M. Oppus, Ant colony Optimization for Image Edge Detection, WSEAS International Conference, 2010.

[8] A. Rezaee, Extracting Edge of Images with Ant Colony, Journal of Electrical Engineering, Vol.59, No.1, 2008, pp. 57-59.

[9] J. Tian, W. Yu, and S. Xie, An Ant Colony Optimization Algorithm for Image Edge Detection, IEEE Congress on Evolutionary Computation, 2008.

[10] H. Nezamabadi-pour, S. Saryazdi, and E. Rashedi, Edge Detection Using Ant Algorithms, Soft Computing, Vol.10, 2006, pp. 623-628.

[11] X. Zhuang, Edge Feature Extraction in Digital Images with the Ant Colony System, IEEE International Conference in Computational Intelligence for Measurement Systems and Applications, 2004.

[12] Y. T. Kim, Contrast Enhancement Using Brightness Preserving Bi-Histogram Equalization, IEEE Trans.,Consumer Electronics, 1997, vol. 43, no. 1, pp. 1-8.

[13]Jing Tian, Weiyu Yu and Shengli Xie, An Ant Colony Optimization Algorithm For Image Edge Detection, IEEE 2008.