Robust Edge Detection Algorithm using Ant Colony System

Inkyeom Kim and Minyoung Yun

Department of Information and Communications Engineering,
Sungkyul University, Anyang, Korea
{kik, alabama }@sungkyul.edu

Abstract. The Ant Colony System (ACS) is easily applicable to the traveling salesman problem (TSP) and it has demonstrated good performance on TSP. Recently, ACS has been emerged as the useful tool for the pattern recognition, feature extraction, and edge detection. The edge detection is widely utilized in the area of document analysis, character recognition, and face recognition. However, the conventional operator-based edge detection approaches require additional post processing steps for the application. In the present study, in order to overcome this shortcoming, we have proposed the new ACS-based edge detection algorithm which has the capabilities to detect finer edges as well as to extract connected edges. The experimental results indicate that this proposed algorithm has the excellent performance in terms of robustness and flexibility.

1 Introduction

The Ant Colony System (ACS) is a meta-heuristic algorithm based on the foraging behavior of ant colonies. Real ants are capable of finding the shortest path from a food source to their nest by exploiting pheromone information. Since its development by Dorigo et al.[1], ACS has been applied to complex combinatorial optimization problems such as the traveling salesman problem (TSP)[2,3], the quadratic assignment problem (QAP), and many discrete optimization problems such as vehicle routing [4], sequential ordering, graph coloring, and routing in communication networks. Recent applications of ACS include pattern recognition, image extraction, and edge detection [5,6,7,11,12].

A significant amount of research is being currently conducted to develop an edge detection algorithm which can be applied to detect and localize the boundaries of objects in an image. Since the detected edges are widely applicable to the areas of document analysis, character recognition, and face recognition, it is crucial for the edge detection algorithm to efficiently and clearly detect the edges. Well-known conventional edge detection operators include the Sobel operator using the gradient-based method, the Laplacian operator based on the second derivation method, and the Canny operator which is the most widely used

© G. Sidorov, M. Aldape, M. Martínez, S. Torres. (Eds.)
Advances in Computer Science and Engineering.
Research in Computing Science 42, 2009, pp. 167-176

Received 06/03/09
Accepted 27/04/09
Final version 08/05/09

edge detection operator. Even though operator based edge detection methods can detect edges clearly, they require an additional step for next-stage image processing.

In the present study, we propose a new ACS-based edge detection algorithm which has the ability to detect finer edges as well as to extract connected edges. Since application of the edge detection algorithm is highly sensitive to the thickness and range of the edges, the edge detection algorithm for actual applications must be flexible in order to accurately represent the thickness and range of the edges. In this regard, the proposed edge detection algorithm is flexible and robust for the application of ACS.

2 Edge Detection Using ACS

Nezamabadi-pour, Saryazdi, and Rashedi[7] applied ACS for detecting edges in digital images. In their approach, the image is considered as a two-dimensional graph where each pixel is denoted as a vertex. Ants move from pixel to pixel and mark the visited pixel with pheromone. In the initial stage, m ants are placed randomly on each pixel. The intensity of the all pixels is set to 0.0001. Ants statistically choose one of their eight-neighboring pixels with the probability described in equation (1). The probability of moving the kth ant from vertex (r, s) to vertex (i, j) is expressed as:

$$p^k_{(r,s),(i,j)} = \begin{cases} \frac{(\tau_{(i,j)})^\alpha (\eta_{(i,j)})^\beta}{\Sigma_u \Sigma_v (\tau_{(u,v)})^\alpha (\eta_{(u,v)})^\beta}, & \text{if } (i,j) \text{ and } (u,v) \in \text{admissible nodes} \\ & r-1 \le i,u \le r+1,\, s-1 \le j,v \le s+1 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The heuristic information $\eta_{i,j}$ of pixel (i, j) is defined by the following formulation.

$$\eta_{i,j} = \frac{1}{I_{Max}} \times Max \begin{bmatrix} |I(i-1,j-1) - I(i+1,j+1)|, \\ |I(i-1,j+1) - I(i+1,j-1)|, \\ |I(i,j-1) - I(i,j+1)|, \\ |I(i-1,j) - I(i+1,j)| \end{bmatrix} \quad (2)$$

After each step, the pheromone is updated by the following equation:

$$\tau_{(i,j)}(new) = (1-\rho)\tau_{(i,j)}(old) + \Delta\tau_{(i,j)} \quad (3)$$

where
$\Delta\tau_{(i,j)} = \Sigma_{k=1}^{m} \Delta\tau^k_{(i,j)}$, and,

$$\Delta\tau^k_{(i,j)} = \begin{cases} \eta_{(i,j)}, & \text{if } \eta_{(i,j)} \ge b \text{ and } k\text{th ant displaces} \\ 0, & \text{otherwise} \end{cases}$$

Here b is a threshold value. If a pixel is not chosen by the ants, its intensity of pheromone decreases exponentially. To avoid stagnation in the searching process, the minimum of pheromone intensity is limited by τ_{min}. Since $\tau_{min} \geq 0$, the probability of choosing a specific pixel can not be zero.

3 Proposed Edge Detection Algorithm

In the proposed edge detection algorithm, edges in the digital image are searched by using the ACS which was developed by Dorigo et. al [2]. The present algorithm detects edges by utilizing the accumulated pheromone on the edge area. The pheromone, τ_{ij}, accumulated at the pixel (i, j) represents the measure of the probability for moving from the current pixel (i, j) to other pixels. Heuristic information of the digital image is obtained from the relationship, $\eta_{ij} = d_{ij}$ and d_{ij} which denotes the intensity difference between the current pixel (i, j) and the neighboring pixels. τ_{ij} and η_{ij} are stored at the matrices of pheromone and heuristic information, respectively. Each ant initially begins the search process at a randomly chosen pixel. Among the unvisited pixels, a next searching pixel is selected according to the values of τ_{ij} and η_{ij} . Each ant updates the pheromone for the visited pixel at every step and all ants update the pheromone one more time for the all the visited pixels if the searching step reaches the predefined number of steps.

If the digital image has a resolution of $M \times N$ pixels, m ants are randomly placed at m pixels in the initial stage. Figure 1 illustrates the edge detection algorithm proposed in this study.

```
algorithm: Proposed ACS for Edge Detection {

  Initialize Data;
  while (not terminate) {
      place m ants at M × N pixels;
      repeat (for each ant)
          apply search construction rule to find edges;
          apply local pheromone updating rule;
      until (construct a solution)
      apply global pheromone updating rule;
      apply evaporation rule;
  }
}
```

Fig. 1. Procedure of the proposed edge detection algorithm

3.1 Search Rule

In this proposed algorithm, when ant k is moved from pixel (i,j), a next pixel (l,h) is chosen by the pseudo-random proportional rule described in the following equation.

$$(l,h)=\begin{cases}\arg\max J, & \text{if } q\leq q_0;\\ \arg\max J, & \text{if } q\geq q_0 \& J\leq q_1;\\ \arg\min J, & \text{if } q\geq q_0 \& J> q_1;\end{cases} \tag{4}$$

where q is a random variable uniformly distributed in $[0.1]$, q_0 $(0\leq q_0\leq 1)$ is a parameter for the random search, and J is a random variable selected by the probability distribution. When the probability for selecting the next pixel exceeds the specified value, a variable, q_1 is applied for searching the homogeneous region. The probability for moving ant k from pixel (i,j) to pixel (l,h) is expressed below.

$$p^k_{(ij),(lh)}=\frac{[\tau_{ij}]^\alpha[\eta_{ij}]^\beta}{\sum_{l\in N_i^k}\sum_{h\in N_j^k}[\tau_{(i+l)(j+h)}]^\alpha[\eta_{(i+l)(j+h)}]^\beta} \quad \text{if } \; l\in N_i^k \& h\in N_j^k \tag{5}$$

Here α and β are parameters for determining the importance of the pheromone τ_{ij} and the relative importance of heuristic information η_{ij}, respectively. N_i^k and N_j^k denote the set of unvisited pixels.

According to equation (5), the selection probability of a pixel (l,h) from a pixel (i,j) is determined by pheromone τ_{ij} and heuristic information η_{ij}. η_{ij} is computed by equation (2) adopted from the previous study [7]. Each ant selects a next visiting pixel (l,h) which has a larger amount of pheromone and the greatest gray-level difference among all pixels in the search. If $\beta=0$, probability for selecting a next visiting pixel depends only on the pheromone level, τ_{ij}. If $\alpha=0$, ants are dependent only on the heuristic information, η_{ij}. To avoid such conditions, α and β must generally satisfy the condition where $\alpha\geq 1$ and $\beta\geq 1$.

3.2 Local Pheromone Update

Whenever an ant visits a pixel (i,j), the pheromone level for the pixel is updated by applying the following equation.

$$\tau_{ij}=(1-\xi)\tau_{ij}+\xi\eta_{ij} \tag{6}$$

Here η_{ij} denotes the difference between the current pixel (i,j) and the neighboring pixels in an 8-neighbor-based search, and ξ is a variable with the range, $0<\xi<1$. According to a Dorigo et al. [4], the best performance is achieved at $\xi=0.1$. By applying equation (6), the pheromone level τ_{ij} at any pixel (i,j)

previously visited by is progressively reduced for each successive visit. Consequently, any previously visited pixel has a much lower probability to be selected by the following ants. Since this procedure increases the probability that ants will select unvisited pixels, it prevents stagnation which is a result of repeated visits to pixels not along the best path.

3.3 Global Pheromone Update

After each ant completes the specified steps, the pheromone levels for a set of visited pixels H^{sp} are updated by the following equation.

$$\tau_{ij} = (1-\rho)\tau_{ij} + \rho\Delta\tau_{ij}^{sp}, \qquad \forall(i,j) \in H^{sp} \tag{7}$$

where $\Delta\tau_{ij}^{sp}$ is the amount of pheromone to be added to pixel (i,j) and it is defined below as:

$$\Delta\tau_{ij}^{sp} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}, \tag{8}$$

$$\Delta\tau_{ij}^{k} = \begin{cases} \eta_{ij}, & \text{if } (i,j) \text{ visited by ant } k \\ 0, & \text{otherwise} \end{cases}$$

Here $\Delta\tau_{ij}^{k}$ represents the heuristic information for a pixel (i,j) visited by an ant k and the parameter ρ denotes the pheromone evaporation rate. In general, the best performance is obtained when $\rho = 0.1$.

3.4 Pheromone Evaporation

Immediately after having visited a pixel (i,j) during the edge detection process, the pheromone is reduced by the local pheromone update rule. On the other hand, the pheromone levels for all pixels visited by an ant are increased according to the global pheromone update rule. Even though the pheromone is deposited and reduced repeatedly due to global and local updates during the search process, it is still possible that pheromone at the certain pixels could be gradually accumulated over time due to the increased searching activity of ants. This accumulation of pheromone could cause stagnation. This situation is resolved by applying pheromone evaporation over the entire search range. The pheromone evaporation effect is imposed by the following equation.

$$\tau_{ij} = (1-\rho)\tau_{ij}, \qquad \forall(i,j) \in I \tag{9}$$

Here I is a total image and evaporation rate ρ has a value 0.1.

4 Experimental Results and Discussion

The experiments on the proposed algorithm are performed on an Enterprise RedHat 2.1. For each test, we chose the parameters which yielded the optimal solution in the previous experiments [2,3,8]. The optimal values of the special parameters for processing the digital images are determined through extensive experiments. In the present study, the problem parameters are:

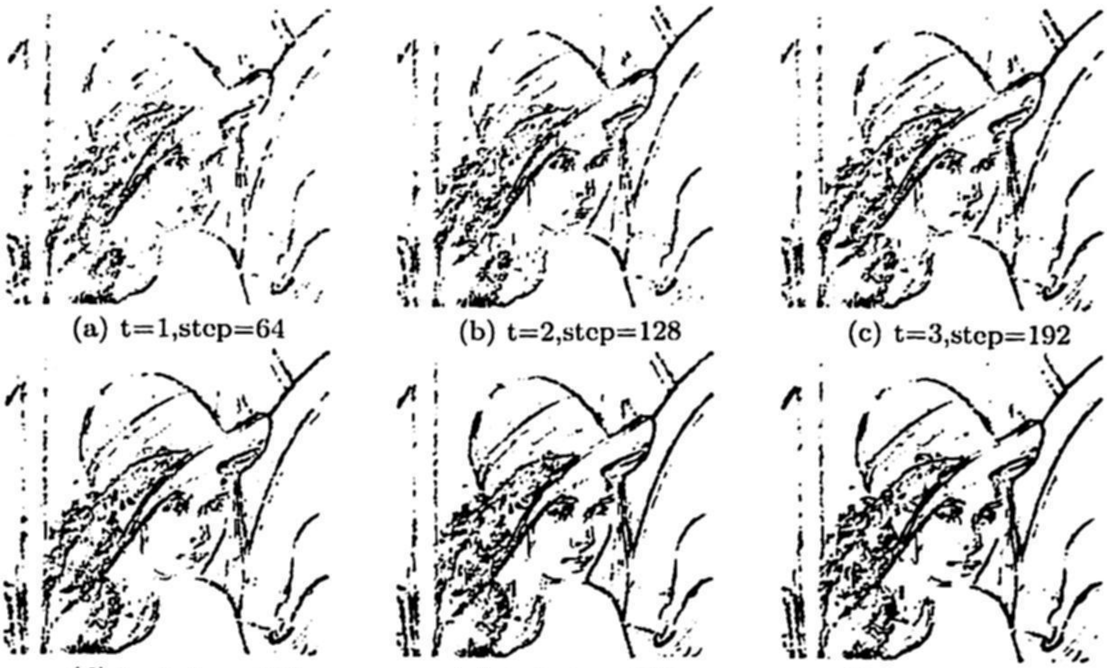
$$\xi = 0.1, \rho = 0.1, \alpha = 2, \beta = 3, q_0 = 0.9, q_1 = 0.25, m = 1000, SR = 64.$$

The initial value of pheromone is set to $\tau_0 = 1/(SR \times I_{max})$ and SR denotes one cycle with 64 steps. The parameter t represents the working time which is required to complete a cycle. After completing a search process of one cycle (64 steps), t is increased by increments of 1 and this search process is continuously performed until t reaches 100.

Since the pheromone levels for all pixels are identical at the initial state, the edges are not clearly detected at the early stage of the searching process. It is assumed that the detected edges will become progressively clear over time. In the moving criteria based on ACS, ants move to the next pixel by using the heuristic information. However, if the moving criterion is determined only by the heuristic information, well detected edges could not be obtained for the general situations. In order to circumvent this drawback, the present approach is set to the search point by utilizing q_0 and q_1 appeared in equation (4). If only q_1 is used for the search criterion, it is quite possible for the ants to search only for background area. However, if q_1 is simultaneously used with q_0 supporting the randomness, the quality of detected edges can be substantially improved.

Figure 2 shows the resulting edge detection images from a search by 1000 ants from 1 to 16 cycles. It can be clearly seen that the fundamental edges of the image are realistically detected. Since the background area is searched only once, the background image is removed due to the pheromone evaporation process. On the other hand, a number of ants visit the edge region where the pheromone is gradually accumulated at the proximity of the edge area. Even if the certain edges are visible in the early stage of search, these edges could diminish due to the pheromone evaporation during the search process. As shown in Figure 2(e) and (f), the experimentally captured images at $t = 8$ and $t = 16$ are nearly identical except for the ridge of nose. This and other differences occur due to pheromone deposition and evaporation during the search. Moreover, the noticeable differences are not observed for the detected edges at $t \geq 10$ and $t = 8$. If the edge detection process relies heavily on the heuristic information, it is quite possible to lose the information for the much finer edges. In the generalized ACS algorithm, even if the parameter β influences the heuristic information, the overall results are not affected by β. However, in the digital image, if β becomes large, stagnation could occur because ants cannot search for the proper edges.

Figure 3 shows the enlarged results obtained for three different values of α and β. Figure 3(a) displays the result when $\alpha = 1, \beta = 3$. At this value, the



(a) t=1,step=64 (b) t=2,step=128 (c) t=3,step=192
(d) t=4,step=256 (e) t=8,step=512 (f) t=16,step=1024

Fig. 2. Results of edge detection by cycle t

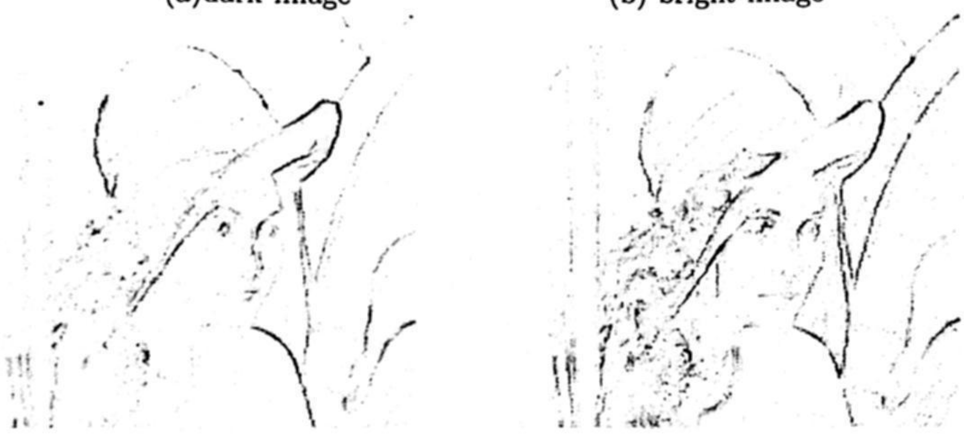
edge detection process is highly dependent on the heuristic information and the corresponding search results include many edges. However, many edges involved in the search results are not always meaningful. Figures 3(b) and 3(c) presents the results obtained for $\alpha = 3, \beta = 2$ and $\alpha = 2, \beta = 3$. As indicated in Figures 3(b) and 3(c), even if the heuristic information is very influential at the large values of β, the edges could be adequately detected by by using an optimal value for α. In terms of edge detection, noticeable differences exist around at the eyes and the ridge of nose. When α is larger than β, the edges are weakly detected on the ridge of the nose and the detected edges close to the nose are widely spread apart, as presented in Figure 3(b). The wide spacing of edges is mainly due to the fact that pheromone levels have a greater influence than the heuristic information on search behavior when α is larger than β. In this study, the optimum values for parameters α and β are obtained by comparing the experimental result having the best edge detection with the edge information of the original image. The experimental results indicate that $\alpha = 2$ and $\beta = 3$ produce the best performance for the edge detection as shown in the Figure 3.

Figure 4 shows the robustness of the proposed method for edge detection. It is shown the results of the detected edges in darker and brighter image than original Lena. In dark image, the overall results is good except the around of the shawl which has so many black region. In case of brighter image, it is shown that more specific edges were detected. Because of the high gray level, it is detected finer edges than in dark image in the shawl around. However, the detected edges compared the original Lena image with the dark and bright images show satisfied results.

(a) alpha=1 beta=3 (b) alpha=3 beta=2 (c) alpha=2 beta=3

Fig. 3. Edge detection results for various values of α and β

(a)dark image (b) bright image



(c) Edge Result of dark image (d) Edge Result of bright image

Fig. 4. Results of edge detection in dark and bright Image

(a) Sobel (b) Laplacian (c) Proposed

Fig. 5. Enlarged results for comparing edge detection

(a) Lena (b) cameraman (c) face

Fig. 6. Results of edge detection using the proposed algorithm

It is compared proposed method with famous edge detection methods of Sobel and Laplacian. It is used 3x3 operators for Sobel and Laplacian. Figure 5 show the results of the enlarged edges of the Lena by Sobel, Laplacian and proposed edge detection method. The result of Sobel show too thick edges. Laplacian presents finer thin edges but edges is spread and not connected. However, the results of the proposed methods are shown finer thin and connected edges. The proposed method presents a robust and flexible results for the edge detection.

Finally, the proposed algorithm is applied to capture the images of Lena, a Cameramen, and a Face. As displayed in Figure 6, these experimental results clearly indicate that the present ACS-based algorithm is quite capable of detecting the edges of various complex images when the proper values for α and β are utilized. Moreover, in the case of Lena, as using the combination of q_0 and q_1, the proposed algorithm successfully detects the finer edges around her eyes and the shawl that were not captured in a previous study [7].

5 Conclusion

In this study, we have proposed an ACS-based algorithm which can accurately detect the edges of the digital images. In the conventional method for the image processing, edge detection is performed for fixed values for the image pixels or by using the known operators. Thus, the conventional operator-based edge detection approaches require additional post processing steps for image processing applications. In order to overcome this shortcoming, we have proposed a new ACS-based edge detection algorithm which has the capabilities to detect finer edges as well as to extract connected edges. The proposed algorithm is able to detect the edges with the combination of pheromone and heuristic information which are produced by ants in random fashion. In the present edge detection procedure, the gray levels around edges in the image are gradually changed, because gray levels are not obtained from any criterion based on the fixed values. The experimental results clearly indicate that the present ACS-based algorithm utilizing the proper values of parameters, q_0 and q_1, is quite capable of detecting edges of complex images. Moreover, in the case of Lena, the proposed algorithm successfully detects the finer edges on eyes and the shawl which were not captured in the previous study. The developed ACS-based algorithm can also be used for other image processes including image segmentation and image recognition.

References

1. M. Dorigo, V. Maniezzo and A. Colorni. The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 26(1) : 1-13, 1996.
2. M. Dorigo and L. M. Gambardella. Ant Colonies for the Travelling Salesman Problem. *BioSystems*, 43:73-81, 1997.
3. M. Yun and I. Kim. Improved Ant Colony System Using Subpath Information for the Traveling Salesman Problem. *Research on Computer Science*, Vol. 16, pp. 185-194, 2005.
4. M. Dorigo, G. D. Caro and L. M. Gambardella. Ant Algorithms for Discrete Optimization. *Artificial Life*, 5(3):137-172, 1999.
5. V. Ramos and F. Almeida. Artificial ant colonies in digital image habitats: a mass behaviour effect study on pattern recognition. *Proceedings of ANTS'2000 - International workshop on ant algorithms*, 113-116, 2000.
6. V. Ramos, F. Muge and P. Pina. Self-organized data and image retrieval as a consequence of inter-dynamic synergistic relationships in artificial ant colonies. *Hybrid Intelligence Systems*, 87, 2002.
7. H. Nezamadadi-pour, S. Saryazdi and E. Rashedi. Edge detection using ant algorithm. *Soft Computing*, August, 2005.
8. R. Gonzales and R. E. Woods. Digital Image Processing. Second Edition, Prentice Hall, 2002.
9. M. Dorigo and T. Stutzle. Ant Colony Optimization. MIT Press, 2003.
10. E. Bonabeau, M. Dorigo and G. Theraulaz. Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, 1999.
11. A. Rezaee. Extracting Edge of Images with Ant Colony. *Journal of Electrical Engineering*, 59(1):57-59, 2008.
12. Y. Yu and L. Guo. Ant Colony search for Edge Extraction in Noise Image. *Journal of Electronics and Information Technology*, 30 (6): 1271-1275, 2008.