Image Edge Detection Using Variation-Adaptive Ant Colony Optimization

Jing Tian[1], Weiyu Yu[2], Li Chen[3], and Lihong Ma[4]

[1] School of Computer Science and Technology, Wuhan University of Science and Technology, P.R. China, 430081
eejtian@gmail.com
[2] School of Electronic and Information Engineering, South China University of Technology, Guangzhou, P.R. China, 510641
yuweiyu@scut.edu.cn
[3] School of Computer Science and Technology, Wuhan University of Science and Technology, P.R. China, 430081
chenli@ieee.org
[4] Guangdong Key Lab of Wireless Network and Terminal, School of Electronic and Information Engineering, South China University of Technology, Guangzhou, P.R. China, 510641
eelhma@scut.edu.cn

Abstract. *Ant colony optimization* (ACO) is an optimization algorithm inspired by the natural collective behavior of ant species. The ACO technique is exploited in this paper to develop a novel image edge detection approach. The proposed approach is able to establish a pheromone matrix that represents the edge presented at each pixel position of the image, according to the movements of a number of ants which are dispatched to move on the image. Furthermore, the movements of ants are driven by the local variation of the image's intensity values. Extensive experimental results are provided to demonstrate the superior performance of the proposed approach.

1 Introduction

Ant colony optimization (ACO) is a nature-inspired optimization algorithm [8] motivated by the natural collective behavior of real-world ant colonies. The major collective behavior is the foraging behavior that guides ants on short paths to their food sources, since ants can deposit pheromone on the ground in order to mark some favorable path that should be followed by other members of the colony. Inspired by this, the first ACO algorithm, called the *ant system*, was proposed by Dorigo *et al.* [7]. Since then, a number of ACO algorithms have been developed [3], such as the *Max-Min ant system* [16] and the *ant colony system* [5]. The ACO technique has been widely applied to tackle various real-world applications [2, 4, 6, 19].

Image edge extraction aims to extract the edge presented in the image, which is crucial to understand the image's content [9]. Conventional image edge detection algorithms usually perform a linear filtering operation (or with a smoothing

N.T. Nguyen (Ed.): Transactions on CCI V, LNCS 6910, pp. 27–40, 2011.

pre-processing operation to remove noise from the image) on the image [9], such as *Sobel* and *Canny* operators. These operators yield low computational load and are suitable for real-time applications. However, these operators usually result in artifacts (e.g., broken edges) in the resulted edge image [9]. To overcome the above drawback, inspired by natural behavior of ants, various ant-based image edge detection methods have been developed [11, 13, 20]. Zhuang [20] proposed to utilize the perceptual graph to represent the relationship among neighboring image pixels, then use the ant colony system to build up the perceptual graph. Nezamabadi-Pour *et al.* [13] proposed to use the ant system to detect edges from images by formulating the image as a directed graph. Lu and Chen [11] proposed to use the ACO technique as a post-processing to compensate broken edges, which are usually incurred in the conventional image edge detection algorithms.

The ACO technique is exploited in this paper to develop a new image edge detection approach. The contribution of this paper is highlighted as follows. The proposed approach aims to establish a pheromone matrix, whose entries represent the edge at each pixel location of the image, by exploiting the ant colony system. The ant colony evolves on the image via moving to adjacent pixels according to the local features of the image. In the context of image edge extraction, the local feature used in the propose approach is the local variation of the image's intensity values (the more contrasted the most favorable a region is).

It is important to note that there are fundamental differences between our proposed approach and the conventional ones [1, 11, 13, 20]. First, our proposed approach exploits the pheromone that is established by ant colony to extract the edge from images; while Zhuang's algorithm [20] exploits a perceptual graph to represent the edge and further applies the ACO technique to construct this graph. Second, our proposed approach exploits the *ant colony system* [5]; on the contrary, Nezamabadi-Pour *et al.*'s method [13] applies the *ant system* [7]. It has been shown that the above fundamental difference (ant colony system *v.s.* ant system) is crucial to the respective designed ACO-based algorithms [3]. Third, ACO is exploited to 'directly' extract the edge in our proposed method, in contrast to that the ACO technique serves as a 'post-processing' in [11] to enhance the edge that has already been extracted by conventional edge detection algorithms.

The paper is organized as follows. An ACO-based image edge detection approach is proposed in Section 2. Extensive experimental results and discussions are presented in Section 3. Finally, Section 4 concludes this paper.

2 Proposed ACO-Based Image Edge Detection Approach

The proposed approach aims to utilize a number of ants to evolve on a 2-D image for constructing a pheromone matrix, each entry of which represents the edge at each pixel location of the image. Furthermore, the movements of the ants are steered by the local variation of the image's intensity values. That is, the ants prefer to move towards positions with larger variations (i.e., edge region).

The proposed approach starts from the *initialization process* and runs for N iterations to construct the pheromone matrix by iteratively performing both the *construction process*, which constructs the pheromone matrix, and the *update process*, which updates the pheromone matrix. Finally, the *decision process* is performed to determine the edge by applying a binary thresholding on the constructed pheromone matrix. Each of the above four process is presented in detail as follows.

2.1 Initialization Process

Totally K ants are randomly assigned on an image $\mathbf{I}$ with a size of $M_1 \times M_2$, each pixel of which can be viewed as a node. The initial value of each component of the pheromone matrix $\boldsymbol{\tau}^{(0)}$ is set to be a constant τ_{init}.

2.2 Construction Process

At the n-th construction-step, one ant is randomly selected from the above-mentioned total K ants, and this ant will consecutively move on the image for L movement-steps. This ant moves from the node (l, m) to its neighboring node (i, j) according to a transition probability, which is defined as

$$p^{(n)}_{(l,m),(i,j)} = \frac{\left(\tau^{(n-1)}_{i,j}\right)^{\alpha} (\eta_{i,j})^{\beta}}{\sum_{(s,q)\in\Omega_{(l,m)}} \left(\tau^{(n-1)}_{s,q}\right)^{\alpha} (\eta_{s,q})^{\beta}}, \tag{1}$$

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

There are two crucial issues in the construction process. The first issue is the determination of the heuristic information $\eta_{i,j}$ in (1). In this paper, it is proposed to be determined by the local statistics at the pixel position (i, j) as

$$\eta_{i,j} = \frac{1}{Z} V_c(I_{i,j}), \tag{2}$$

where Z is a normalization factor as

$$Z = \sum_{i=1:M_1} \sum_{j=1:M_2} V_c(I_{i,j}), \tag{3}$$

$I_{i,j}$ is the intensity value of the pixel at the position (i, j) of the image $\mathbf{I}$, the function $V_c(I_{i,j})$ is a function of a local group of pixels c, and its value depends on the variation of image's intensity values on the clique c (as shown in Figure 1). More specifically, for the pixel $I_{i,j}$ under consideration, the function $V_c(I_{i,j})$ is

$$\begin{aligned} V_c(I_{i,j}) = f\,(&|I_{i-2,j-1} - I_{i+2,j+1}| + |I_{i-2,j+1} - I_{i+2,j-1}| + |I_{i-1,j-2} - I_{i+1,j+2}| \\ &+|I_{i-1,j-1} - I_{i+1,j+1}| + |I_{i-1,j} - I_{i+1,j}| + |I_{i-1,j+1} - I_{i-1,j-1}| \\ &+|I_{i-1,j+2} - I_{i-1,j-2}| + |I_{i,j-1} - I_{i,j+1}|\,). \end{aligned} \tag{4}$$

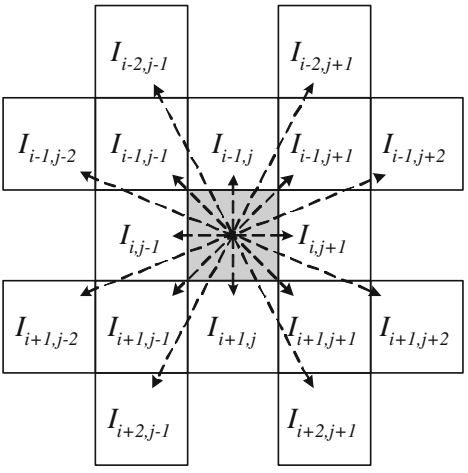


Fig. 1. A local configuration at the pixel position $I_{i,j}$ for computing the variation $V_c(I_{i,j})$ defined in (4). The pixel $I_{i,j}$ is marked as gray square.

The key issue is to establish the function $f(\cdot)$ used in (4), which is defined as

$$f(x) = \begin{cases} \sin\left(\frac{\pi x}{2\lambda}\right) & 0 \le x \le \lambda; \\ 0 & \text{else}, \end{cases} \tag{5}$$

where the parameter λ adjusts the function's shape. The discussion of various functions is provided in Section 4.2.1.

2.3 Update Process

The proposed approach performs two updates operations for updating the pheromone matrix.

– The first update is performed after the movement of each ant within each construction-step. Each component of the pheromone matrix is updated according to

$$\tau_{i,j}^{(n-1)} \leftarrow \begin{cases} (1-\rho)\cdot\tau_{i,j}^{(n-1)} + \rho\cdot\Delta_{i,j}^{(k)}, & \text{if } (i,j) \text{ is visited by} \\ & \text{the current } k\text{-th ant}; \\ \tau_{i,j}^{(n-1)}, & \text{otherwise.} \end{cases} \tag{6}$$

where ρ is evaporation rate, $\Delta_{i,j}^{(k)}$ is determined by the heuristic matrix; that is, $\Delta_{i,j}^{(k)} = \eta_{i,j}$.

– The second update is carried out after the movement of all ants within each construction-step according to

$$\boldsymbol{\tau}^{(n)} = (1-\psi)\cdot\boldsymbol{\tau}^{(n-1)} + \psi\cdot\boldsymbol{\tau}^{(0)}, \tag{7}$$

where ψ is pheromone decay coefficient.

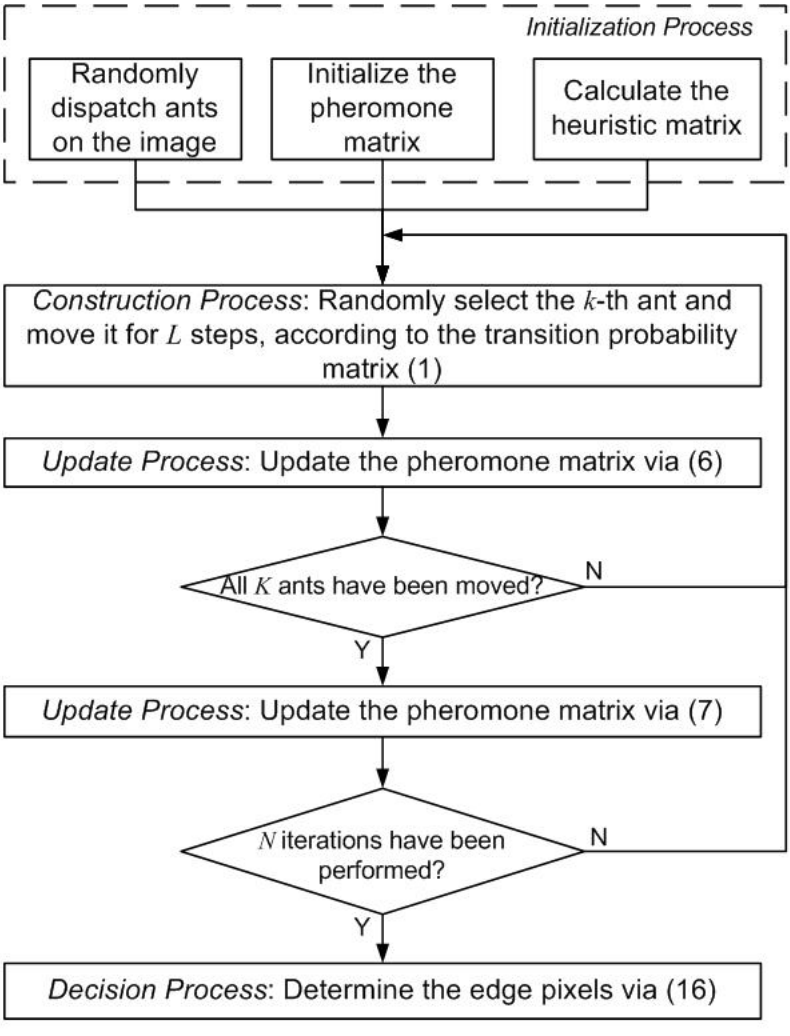


Fig. 2. A summary of the implementation of the proposed approach

2.4 Decision Process

In this step, a binary decision is made at each pixel location to determine whether it is edge or not, by applying a threshold T on the final pheromone matrix $\boldsymbol{\tau}^{(N)}$. This threshold can be either manually determined by users or automatically determined by any binary thresholding algorithm (e.g., Otsu's algorithm in [14]). In this paper, the above-mentioned T is proposed to be adaptively computed based on the method developed in [14] as follows.

The initial threshold $T^{(0)}$ is selected as the mean value of the pheromone matrix. Next, the entries of the pheromone matrix is classified into two categories

(a) (b)

Fig. 3. Test images used in this paper: (a) *Camera* (128×128) and (b) *House* (128×128)

according to the criterion that its value is lower than $T^{(0)}$ or larger than $T^{(0)}$. Then the new threshold is computed as the average of two mean values of each of above two categories. The above process is repeated until the threshold value does not change any more (in terms of a user-defined tolerance ϵ).

2.5 The Summary of the Proposed Approach

A summary of the implementation of the proposed image edge extraction approach is presented in Figure 2.

3 Experimental Results

3.1 Experimental Setup and Implementation

Experiments are conducted to evaluate the performance of the proposed approach using two test images, *Camera* and *House*, which are shown in Figure 3. Furthermore, various parameters of the proposed approach are set as follows. The total number of ants K is set to be $\lfloor\sqrt{M_1 \times M_2}\rfloor$, where the function $\lfloor x \rfloor$ represents the highest integer value that is smaller than or equals to x. The initial value of each component of the pheromone matrix τ_{init} is set to be 0.0001. The weighting factors of the pheromone and the heuristic information in (1) are set to be $\alpha = 10$ and $\beta = 0.01$, respectively. The permissable ant's movement range in (1) is selected to be 8-connectivity neighborhood. The adjusting factor of the function in (5) is $\lambda = 1$. The evaporation rate in (6) ρ is 0.1. Total number of ant's movement-steps within each construction-step L is 40. The pheromone decay coefficient in (7) $\psi = 0.05$. The user-defined tolerance value used in the decision process of the proposed method $\epsilon = 0.1$. Total number of construction-steps N is experimentally selected to be 4.

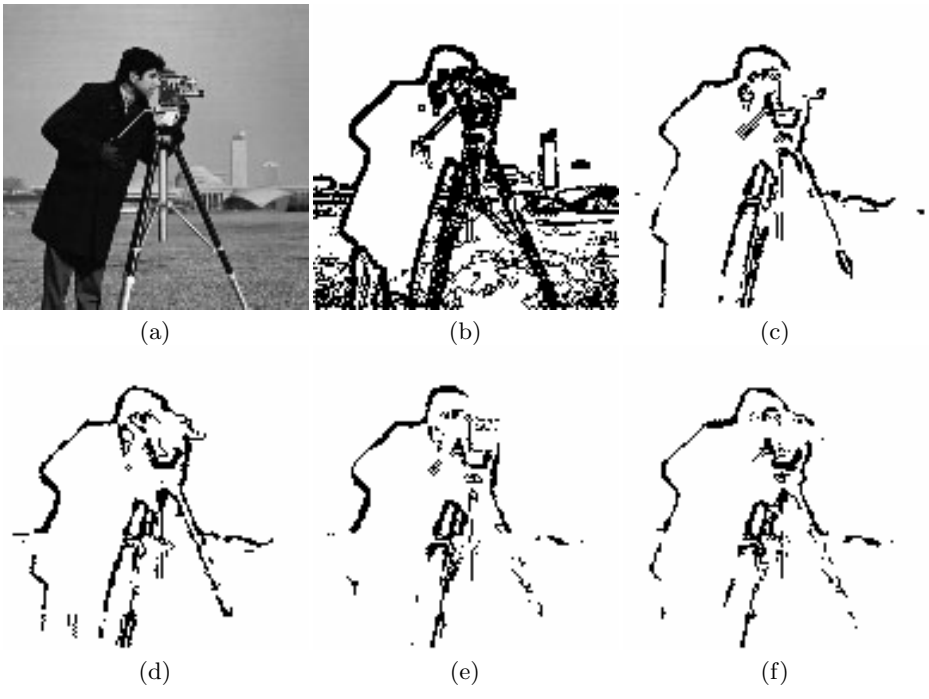
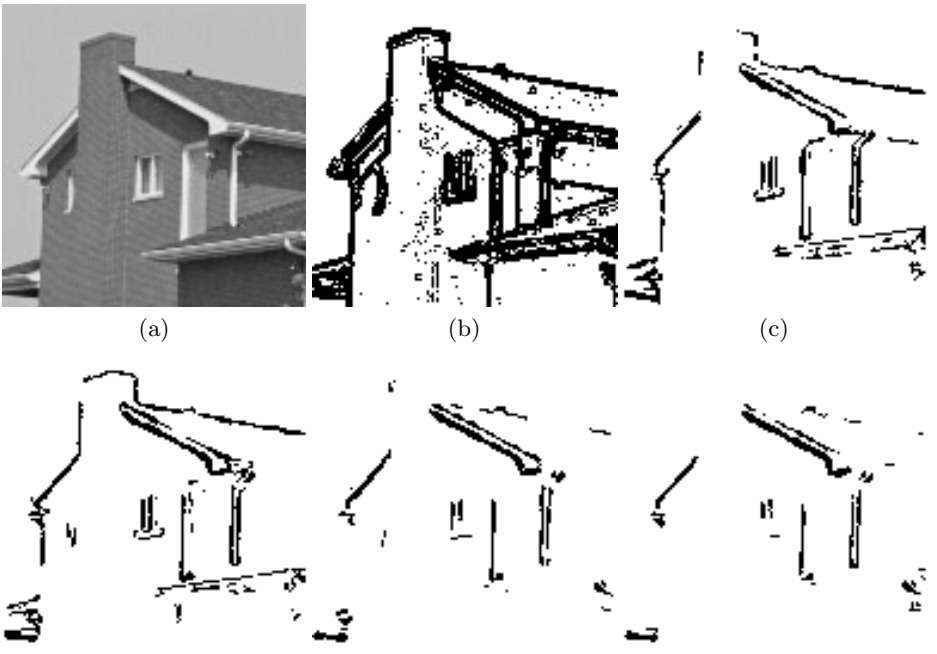


Fig. 4. Functions used in establishing the heuristic information matrix ($f(\cdot)$ in (4)): (a) the original *Camera* image; (b) Nezamabadi-Pour *et al.*'s method [13]; (c)-(f) proposed algorithm with the incorporation of the functions defined in (5), (8)-(10), respectively.

3.2 Experimental Results and Discussions

Experimental results are provided to compare the proposed approach with the state-of-the-art method developed in the literature, e.g., Nezamabadi-Pour *et al.*'s method [13]. To provide a fair comparison, the morphological thinning operation of [13] is neglected, since it is performed as a post-processing to further refine the edge that is extracted by the ACO technique [13]. As seen from Figures 4 and 5, the proposed provides superior performance to that of Nezamabadi-Pour *et al.*'s method [13], in terms of visual image quality. Furthermore, the determination of several parameters are critical to the performance of the proposed approach; this issue will be discussed in detail as follows.

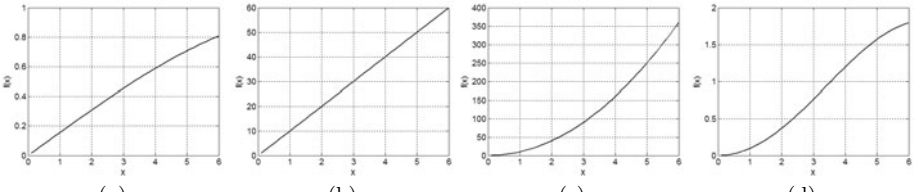
Functions Used in Establishing the Heuristic Information Matrix ($f(\cdot)$ in (4)). To determine the function $f(\cdot)$ in (4), the following three functions could be incorporated into the proposed approach, besides the function (5). All these functions are mathematically expressed as follows and illustrated in Figure 6, respectively.



(a) (b) (c)

(d) (e) (f)

Fig. 5. Functions used in establishing the heuristic information matrix ($f(\cdot)$ in (4)): (a) the original *House* image; (b) Nezamabadi-Pour *et al.*'s method [13]; (c)-(f) proposed algorithm with the incorporation of the functions defined in (5), (8)-(10), respectively



(a) (b) (c) (d)

Fig. 6. Various functions ($\lambda = 10$) used for establishing the heuristic information matrix: (a) the function defined in (5); (b) the function defined in (8); (c) the function defined in (9); and (d) the function defined in (10)

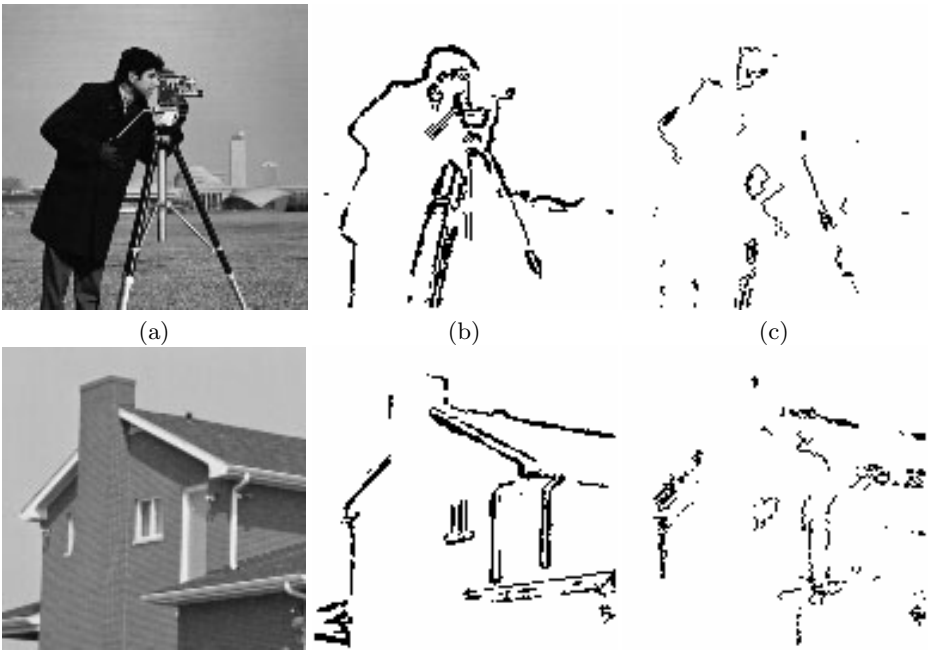


Fig. 7. Weighting factors of the pheromone (α in (1)) and the heuristic information (β in (1)): (a) the original image *Camera*; (b) the proposed algorithm ($\alpha = 10, \beta = 0.01$); (c) the proposed algorithm ($\alpha = 0.01, \beta = 10$); (d) the original image *House*; (e) the proposed algorithm ($\alpha = 10, \beta = 0.01$); and (f) the proposed algorithm ($\alpha = 0.01, \beta = 10$).

$$f(x) = \lambda x, \quad \text{for} \quad x \geq 0, \tag{8}$$

$$f(x) = \lambda x^2, \quad \text{for} \quad x \geq 0, \tag{9}$$

$$f(x) = \begin{cases} \frac{\pi x \sin(\frac{\pi x}{\lambda})}{\lambda} & 0 \leq x \leq \lambda; \\ 0 & \text{else.} \end{cases} \tag{10}$$

The parameter λ in each of above functions (8)-(10) adjusts the functions' respective shapes.

To present how the determination of the heuristic information matrix (i.e., (2)) is crucial to the proposed method, various functions defined in (5), (8)-(10) are individually incorporated into (4) of the proposed approach, and their resulted performances are presented. Figures 4 and 5 present the results of test images *Camera* and *House*, respectively.

Weighting Factors of the Pheromone (α in (1)) and the Heuristic Information (β in (1)). To demonstrate how the weighting factors of the pheromone (i.e., α in (1)) and the heuristic information (i.e., β in (1)) affect the

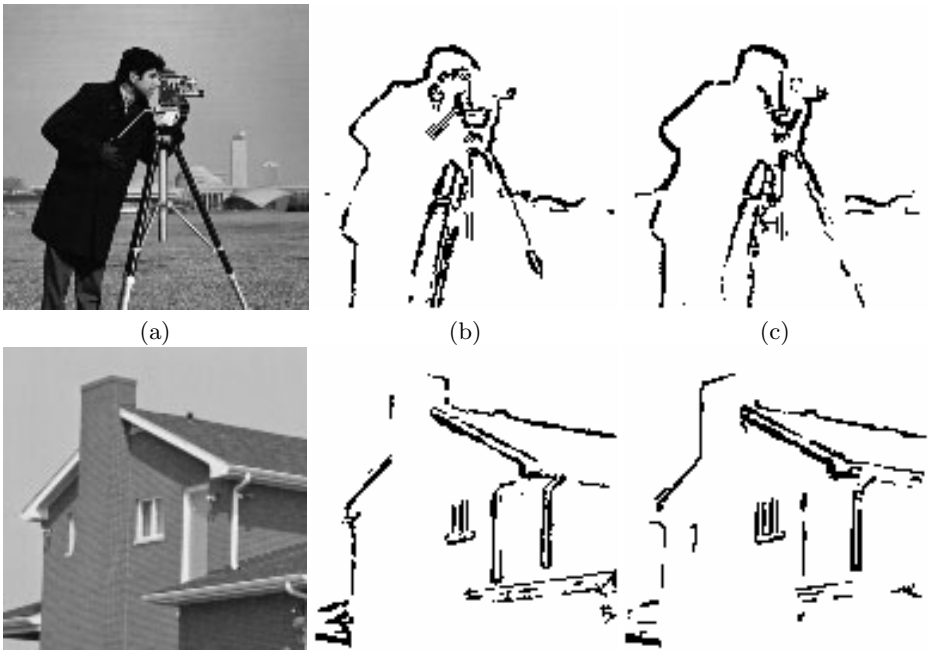
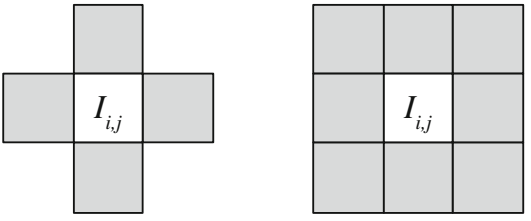


Fig. 8. Permissable range of ant's movement ($\Omega_{(l,m)}$ in (1)): (a) the original image *Camera*; (b)-(c) the proposed algorithm using 8-connectivity neighborhood and 4-connectivity neighborhood, respectively; (d) the original image *House*; (e)-(f) the proposed algorithm using 8-connectivity neighborhood and 4-connectivity neighborhood, respectively.

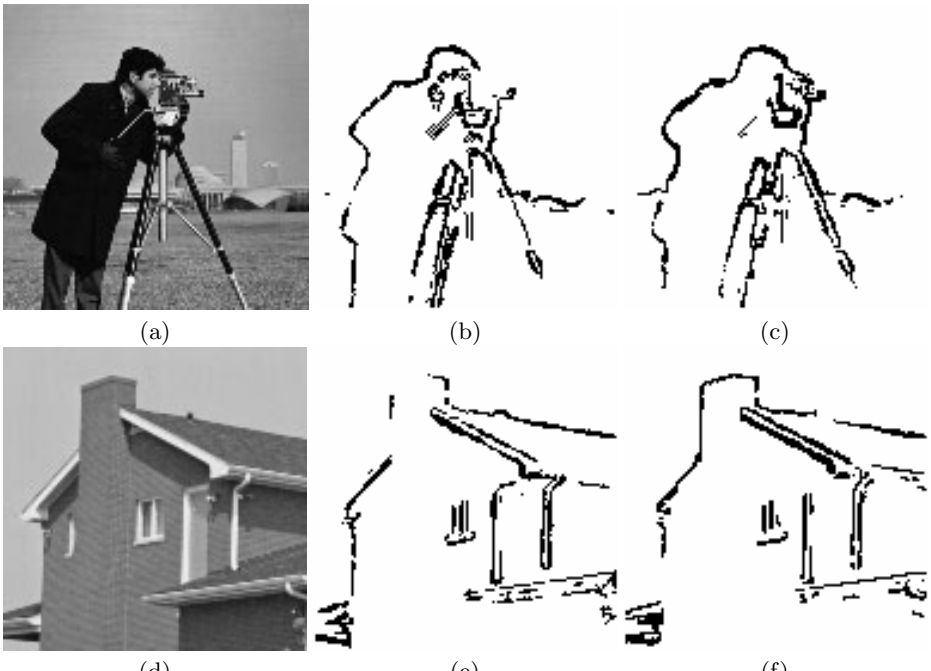
performance of the proposed algorithm, two experiments are conducted using two parameter setups: i) $\alpha = 10, \beta = 0.01$ and ii) $\alpha = 0.01, \beta = 10$. Their respective results are presented in Figure 7. As seen from the Figure 7, a large α value and a small β value will let ants tend to discover edges that have already been discovered by other ants; on the contrary, a small α value and a large β value will let ants tend to discover edges using stochastic movements.

Permissable Range of Ant's Movement ($\Omega_{(l,m)}$ in (1)). The permissable range of the ant's movement (i.e., $\Omega_{(l,m)}$ in (1)) at the position (l, m) could be either the 4-connectivity neighborhood or the 8-connectivity neighborhood. Both of these two cases are demonstrated in Figure 9 and their respective results are shown in Figure 8. As seen from Figure 9, a large permissable movement range will let ants to move to a far position; consequently, much edge will be discovered by ants. On the other hand, a small permissable movement range will let ants to prefer staying in their current positions; consequently, less edge will be discovered by ants.



(a) (b)

Fig. 9. Various permissable range of the ant's movement (marked as gray regions) at the position $I_{i,j}$: (a) 4-connectivity neighborhood; and (b) 8-connectivity neighborhood



(a) (b) (c)

(d) (e) (f)

Fig. 10. Influence of evaporation factor (ρ in (6)): (a) the original image *Camera*; (b) the proposed algorithm ($\rho = 0.1$); (c) the proposed algorithm ($\rho = 0.99$); (d) the original image *House*; (e) the proposed algorithm ($\rho = 0.1$); and (f) the proposed algorithm ($\rho = 0.99$)

Influence of Evaporation Factor (ρ in (6)). Evaporation factor plays a key role to determine the performance of the proposed algorithm. To demonstrate this point, experiments are conducted using different evaporation factors and their results are presented in Figure 10.

Computational Cost. Experiments are conducted to compare the computational cost of the proposed algorithm and that of Nezamabadi-Pour *et al.*'s algorithm [13]. Both of the above two algorithms are implemented using the *Matlab* programming language and run on a PC with a Intel Core^{TM} DUO E6400 2.13 GHz CPU and a 1 GB RAM. Furthermore, both of them are implemented for ten times, then their respective average running times are recorded.

The running times of the proposed approach are 64.97 seconds and 63.73 seconds for the test image *Camera* and *House*, respectively. On the other hand, the running time of Nezamabadi-Pour *et al.*'s algorithm [13] are 53.02 seconds and 53.18 seconds for the above two test images, respectively. Therefore, the proposed algorithm yields slight extra computational load, 22.54% and 19.84% for test images *Camera* and *House*, respectively, compared with that of Nezamabadi-Pour *et al.*'s algorithm [13]. However, this extra computational cost is tolerable, since the visual quality of our result is much better than that of Nezamabadi-Pour *et al.*'s algorithm [13]. On the other hand, since the ACO algorithm is inherently suitable to be parallelized, various parallel ACO algorithms [15, 10] can be exploited to further shorten the running time of the proposed algorithm for future research work.

4 Conclusions

In this paper, an ACO-based image edge detection approach has been developed to exploit the pheromone that is established by the ACO technique to determine the edge presented in the image. Fairly extensive simulations have been conducted to demonstrate the performance of the proposed algorithm using different parameters.

There are few issues of the proposed approach need further investigation. First, images are usually corrupted with noises in image acquisition or image communication. The ACO technique has been reported to be fairly robust to handle the edge detection in noisy image cases [1]. Second, the proposed approach exploits the conventional ACO algorithm [3] that updates the pheromone matrix using the initial pheromone value (see (7)). It would be important to investigate whether such update scheme is efficient. Third, since the edge is very critical to determine image's quality, it would be interesting to see how the edge detected by the proposed approach could benefit other image processing applications, such as image denoising [18, 17] and image saliency detection [12].

Acknowledgement. This work was supported by National Natural Science Foundation of China (60972133); Key Project of Chinese Ministry of

Education (210139); SRF for ROCS, SEM; GDSF Team Project (Grant No. 9351064101000003); the Fundamental Research Funds for the Central Universities (x2dxD2105260), SCUT; Fund of Provincial Key Lab. for Computer Information Processing Tech. (KJS0922).

References

1. Aydın, D.: An efficient ant-based edge detector. In: Nguyen, N.T., Kowalczyk, R. (eds.) Transactions on Computational Collective Intelligence I. LNCS, vol. 6220, pp. 39–55. Springer, Heidelberg (2010)
2. Cordon, O., Herrera, F., Stutzle, T.: Special Issue on Ant Colony Optimization: Models and Applications. Mathware and Soft Computing 9 (December 2002)
3. Dorigo, M., Birattari, M., Stutzle, T.: Ant colony optimization. IEEE Computational Intelligence Magazine 1, 28–39 (2006)
4. Dorigo, M., Caro, G.D., Stutzle, T.: Special Issue on Ant Algorithms. Future Generation Computer Systems 16 (June 2000)
5. Dorigo, M., Gambardella, L.M.: Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE Trans. on Evolutionary Computation 1, 53–66 (1997)
6. Dorigo, M., Gambardella, L.M., Middendorf, M., Stutzle, T.: Special Issue on Ant Colony Optimization. IEEE Transactions on Evolutionary Computation 6 (July 2002)
7. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: Optimization by a colony of cooperating agents. IEEE Trans. on Systems, Man and Cybernetics, Part B 26, 29–41 (1996)
8. Dorigo, M., Thomas, S.: Ant Colony Optimization. MIT Press, Cambridge (2004)
9. Gonzalez, R.C., Woods, R.E.: Digital image processing. Prentice Hall, Harlow (2007)
10. Janson, S., Merkle, D., Middendorf, M.: Parallel ant colony algorithms. In: Alba, E. (ed.) Parallel Metaheuristics: A New Class of Algorithms. Wiley-Interscience, Hoboken (2005)
11. Lu, D.S., Chen, C.C.: Edge detection improvement by ant colony optimization. Pattern Recognition Letters 29, 416–425 (2008)
12. Ma, L., Tian, J., Yu, W.: Visual saliency detection in image using ant colony optimisation and local phase coherence. Electronics Letters 46, 1066–1068 (2010)
13. Nezamabadi-Pour, H., Saryazdi, S., Rashedi, E.: Edge detection using ant algorithms. Soft Computing 10, 623–628 (2006)
14. Otsu, N.: A threshold selection method from gray level histograms. IEEE Trans. Syst., Man, Cybern. 9, 62–66 (1979)
15. Randall, M., Lewis, A.: A parallel implementation of ant colony optimization. Journal of Parallel and Distributed Computing 62, 1421–1432 (2002)
16. Stutzle, T., Holger, H.: Max-Min ant system. Future Generation Computer Systems 16, 889–914 (2000)
17. Tian, J., Chen, L.: Image despeckling using a non-parametric statistical model of wavelet coefficients. Electronics Letters 6 (2011)

18. Tian, J., Yu, W., Ma, L.: Antshrink: Ant colony optimization for image shrinkage. Pattern Recognition Letters (2010)
19. Tian, J., Yu, W., Xie, S.: An ant colony optimization algorithm for image edge detection. In: Proc. IEEE Congress on Evolutionary Computation, Hongkong, China, pp. 751–756 (June 2008)
20. Zhuang, X.: Edge feature extraction in digital images with the ant colony system. In: Proc. IEEE Int. Conf. on Computational Intelligence for Measurement Systems and Applications, pp. 133–136 (July 2004)