



A convenient and robust edge detection method based on ant colony optimization

Xiaochen Liu, Suping Fang*

State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, No. 28 Xianning West Road, Xi'an, Shaanxi 710049, China



ARTICLE INFO

Article history:

Received 17 November 2014

Received in revised form

23 April 2015

Accepted 7 May 2015

Available online 15 May 2015

Keywords:

Edge detection

Ant colony optimization

Ant algorithms

ABSTRACT

Edge detection is usually used as a preprocessing operation in many machine vision industrial applications. Recently, ant colony optimization (ACO) as a relatively new meta-heuristic approach has been used to tackle the edge detection problem. In this work, a convenient and robust method for edge detection based on ACO is proposed, which employs a new heuristic function, adopts a user-defined threshold in pheromone update process and provides a group of suitable parameter values. Experimental results clearly demonstrated the effectiveness of the proposed method, and at the same time, in the presence of noise, the proposed approach outperforms other two ACO-based edge detection techniques and four conventional edge detectors.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Edge detection is an essential preprocessing operation in many machine vision industrial applications [1–3], such as shape recognition, 3D reconstruction, defect detection on mechanical parts, etc. Edges are sets of pixels in the image regions with sharp intensity changes and correspond to visible contour features of objects in an image. Normally, edge detection is a process that inputs a grayscale image and then outputs a binary image to indicate the edges of objects.

Many edge detection methods have been proposed in the last decades. Most of them are based on digital differential methods [4], such as Sobel, Roberts, Laplacian operators, and so on. However these algorithms are quite sensitive to noise, in order to suppress noise, Marr et al. [5] and Canny [6] applied the Gaussian pre-smoother to the image before detecting edges. Unfortunately, this procedure would blur edges while removing noises in the image. Consequently, these approaches would sacrifice the locating accuracy of the detected edges to a certain extent [7], as shown in Fig. 1. To overcome this limitation, edge detection could be formulated as an optimization problem. Ant colony optimization (ACO) [8] as a relatively new optimization approach has been used for edge detection, which could be classified into two categories: direct edge detection [9–18] and broken edge compensation [19,20]. In this work, the proposed method is based on the former, because the latter is used just as a complementary tool to other

edge detectors.

ACO as a swarm intelligence approach has been adopted to directly detect image edges by Zhuang [9,10] since 2004. He used the Ant Colony System (ACS) to build the perceptual graph of images for extracting edge features. Unfortunately, his two methods are only capable of detecting simple edges. After that, Nezamabadi-Pour et al. [11] exploited the Ant System (AS) and applied the directed graph to detect edges. Even though they derived the relationship between image area and the number of ants, they did not use other information about the image for more parameter setting. In practice, the more the adaptive parameters are provided, the more conveniently the proposed method can be used. In this work, we employed three adaptive parameters. Tian et al. [12] also adopted the ACS and proposed the method of computing adaptive threshold to tackle the edge detection problem. Similarly, Jevtic et al. [13] first used multiscale adaptive gain for image contrast enhancement, then applied the ACS to detect image edges. Since then many adaptive thresholding methods [14–18] have been presented. But in fact, the ACO meta-heuristic approach as a swarm intelligence technique is inherently adaptive, because this technique is a collective behavior of decentralized, self-organized agents in a swarm. Therefore, in this work, a user-defined threshold in the pheromone update process is adopted to take advantage of this feature of ACO, and noises can be suppressed effectively by adjusting the user-defined threshold.

In addition, the heuristic information matrices applied to ACO-based edge detection could be generally divided into two types: one is proposed by Nezamabadi-Pour et al., which was used in literatures [11,13,15,16]; the other adopted in [12,14,17,18] was

* Corresponding author.

E-mail address: spfang@mail.xjtu.edu.cn (S. Fang).

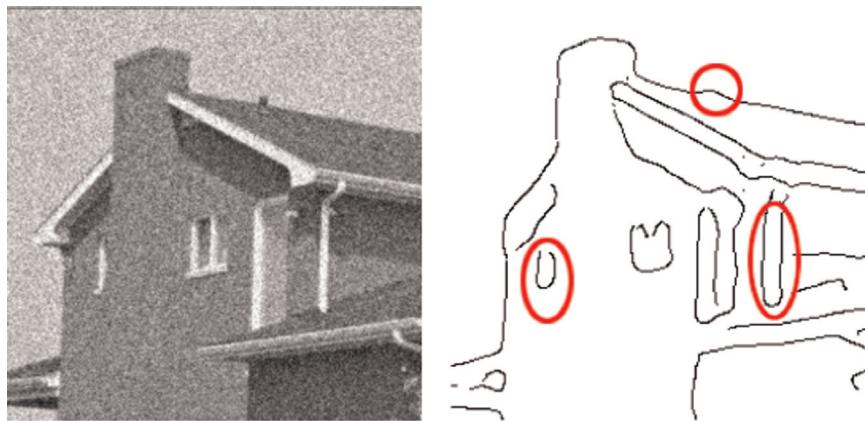


Fig. 1. (Left) Noisy House image, 256 × 256 pixels and (Right) result of Canny edge detector.

devised by Tian et al. On the basis of the two types of heuristic information matrices, a new heuristic information matrix (i.e. Eq. (5) in Section 3) is proposed in this work to improve the gradient response on the edge.

The remaining sections of this paper are organized as follows. First, the main differences between AS and ACS and the implementation of AS algorithm are briefly described in Section 2. Then, the proposed AS-based method for detecting image edges is presented in Section 3. Next, the parameter setting of the proposed approach and the experimental results of performance comparison are given in Section 4. Finally, the conclusions are drawn in Section 5.

2. Ant colony optimization

The basic principle of ACO has been described in more detail in the literature [8], hence we will not repeat it in this section. Since both AS and ACS have been used to detect image edges, we need to simply introduce the differences between them. Besides, we also briefly describe the implementation of AS algorithm.

2.1. Differences between AS and ACS

Several ACO algorithms have been proposed in the literature [8]. As stated in Section 1, we know that two ACO algorithms have been successfully applied to detect image edges: one is the Ant System (AS); the other is the Ant Colony System (ACS). There are two main differences between AS and ACS: **First**, the pheromone update process is different, AS performs one-time updating operation while ACS performs two times. **Second**, the node transition rule of ants is different, AS introduces a stochastic mechanism whereas ACS adopts a pseudo-random-proportional rule. See Section 3.1.1.3 of the literature [21] for more details of their differences. Due to more simplified pheromone update operation in AS algorithm, it costs less running time during the optimization process. Hence, taking the efficiency of both algorithms into account, AS algorithm is employed in the proposed approach. In the next section, the implementation of AS algorithm will be briefly described.

2.2. Implementation of AS algorithm

AS algorithm contains the following four steps:

1. Initialization: a certain number of ants are placed at randomly chosen nodes.
2. Node transition rule: the k th ant moves from node i to node j

according to a probability given by

$$p_{ij}^k = \begin{cases} \frac{(\tau_{ij})^\alpha (\eta_{ij})^\beta}{\sum_{h \in \Omega} (\tau_{ih})^\alpha (\eta_{ih})^\beta} & \text{if } j \in \Omega, \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

where τ_{ij} and η_{ij} are the pheromone value and the heuristic information value for the k th ant moving from node i to node j , respectively; α is the influence coefficient for τ , and β is the influence coefficient for η ; h is a node not visited by the k th ant; and Ω represents the permissible ant's movement range.

3. Pheromone updating rule: an iteration of the AS algorithm is completed when all the ants have built a solution in the iteration. At the end of each iteration, the pheromone τ_{ij} is updated by

$$\tau_{ij}(\text{new}) = (1 - \rho)\tau_{ij}(\text{old}) + \sum_{k=1}^m \Delta \tau_{ij}^k, \quad (2)$$

where ρ is the pheromone evaporation rate; m indicates the number of ants; and $\Delta \tau_{ij}^k$ represents the amount of pheromone deposited on edge (i, j) by the k th ant

$$\Delta \tau_{ij}^k = \begin{cases} \frac{f_k}{Q} & \text{if the } k\text{th ant used edge } (i, j) \text{ in its tour,} \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

where f_k is the fitness value of the solution found by the k th ant, and Q is a constant.

4. Termination criterion: When the predefined number of iterations is reached or the acceptable solution is found, the algorithm will terminate.

3. The proposed method

ACO-based edge detection methods could transform the image intensity values into the pheromone values left by artificial ants in the images. According to these deposited pheromone values, the image edges could be detected.

3.1. Overview of the proposed approach

First, the input gray-level image is regarded as a two-dimensional graph; a pixel is equivalent to a node of the two-dimensional graph. Then, m artificial ants are randomly placed in the input image before starting iteration. Next, one ant is randomly

chosen as the k th ant, it moves in the image pixel by pixel in a fixed number of steps according to a node transition rule (e.g. Eq. (4)), and deposits pheromone at the pixels it just passed. After all the m ants go through their paths, the pheromone is updated by a pheromone updating rule (e.g. Eq. (6)). The algorithm terminates when a specific number of iterations is reached. Finally, the image edges are located at the nodes with stronger pheromone concentrations in the two-dimensional graph.

3.2. Implementation of the proposed approach

The proposed approach is mainly composed of four steps: The initialization process, the construction process of node transition rule, the pheromone update process, and the termination criterion. Each of these processes is presented in detail as follows:

1. Initialization process: the m artificial ants are randomly placed in the image I . The initial pheromone value of each pixel is set to 0.0001.
2. Construction process of node transition rule: In the n th construction step, the k th ant is chosen from the m artificial ants through a stochastic mechanism, and this ant will continuously move from one node (r, s) to its neighborhood node (i, j) in the image I , according to the node transition rule given by

$$p_{(r,s),(i,j)}^n = \begin{cases} \frac{(\tau_{(i,j)}^{(n-1)})^\alpha (\eta_{(i,j)})^\beta}{\sum_{(i,j) \in \Omega_{(r,s)}} (\tau_{(i,j)}^{(n-1)})^\alpha (\eta_{(i,j)})^\beta} & \text{if } (i, j) \in \Omega_{(r,s)} \\ 0 & \text{otherwise,} \end{cases} \quad (4)$$

where $\tau_{(i,j)}^{(n-1)}$ and $\eta_{(i,j)}$ are the pheromone and heuristic information values of node (i, j) , respectively; $\Omega_{(r,s)}$ is the unvisited neighborhood nodes of node (r, s) ; α and β control the influence on pheromone and heuristic information, respectively.

The heuristic information $\eta_{(i,j)}$ plays a very important role in selecting which nodes the ants will visit in the vicinity of their positions. In order to enhance the perceptibility of ants at node (i, j) , a 5×5 structure (as shown in Fig. 2.) is devised. The new heuristic information function at node (i, j) is defined by

$$\begin{aligned} \eta_{(i,j)} &= \frac{1}{I_{max}} \cdot \max \left[|I_{(i-u,j-v)} - I_{(i+u,j+v)}| \right], \quad u \\ &= 0, 1, 2, \quad v \\ &= -2, -1, 0, 1, 2, \end{aligned} \quad (5)$$

where I_{max} is the maximum intensity value of gray-level image I , which is equivalent to a normalization factor; $I_{(i,j)}$ is the intensity value of node (i, j) in image I ; $\max[\bullet]$ is the maximum absolute value of the intensity difference between two nodes with the same color in Fig. 2.

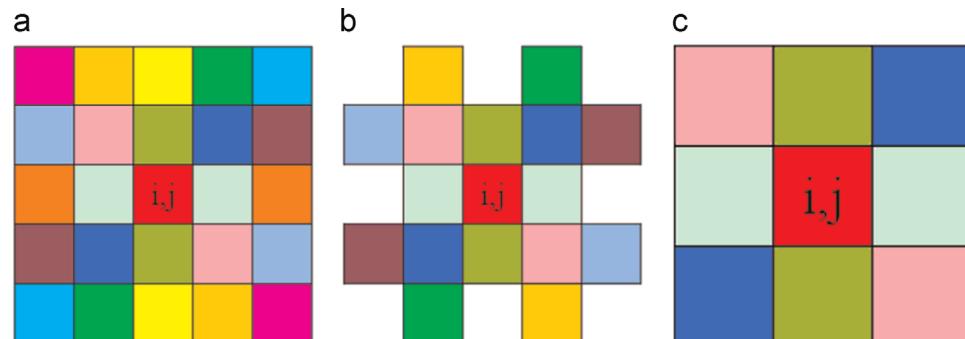


Fig. 2. Three different structures for computing the heuristic information value. (a) Proposed structure; (b) Tian's structure; and (c) Nezamabadi-pour's structure. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

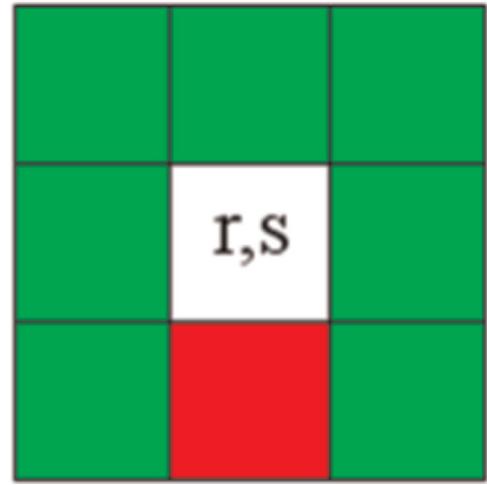


Fig. 3. Admissible range of the ant movement at node (r, s) . (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

Note that the color has no meaning by itself, the use of the colors is to indicate the node position for calculating $\max[\bullet]$ in Eq. (5).

In the proposed approach, an 8-connectivity neighborhood is applied to determine the admissible range of the ant movement at node (r, s) , that is $\Omega_{(r,s)}$ in Eq. (4), as shown in Fig. 3. The green squares are the unvisited neighborhood nodes while ants are located at node (r, s) . The red square represents the visited node in the k th ant's memory. Usually, the memory length of artificial ants l is a user-defined parameter, which is used for better simulating the behavior of real ants.

1. Pheromone update process: after all the m artificial ants moved in each construction process, the pheromone value is updated by the following rule:

$$\tau_{(i,j)}(\text{new}) = (1 - \rho)\tau_{(i,j)}(\text{old}) + \sum_{k=1}^m \Delta\tau_{(i,j)}^k, \quad (6)$$

where ρ is the pheromone evaporation rate, which is used to avoid limitless deposition of the pheromone trails and restrain the ants from choosing the same route (i.e. prevent the stagnation of the algorithm). $\Delta\tau_{ij}^k$ is the pheromone deposited at node (i, j) by the k th ant and is given by

$$\Delta\tau_{(i,j)}^k = \begin{cases} \eta_{(i,j)} & \text{if node } (i, j) \text{ is visited by the } k\text{th ant and } \eta_{(i,j)} \geq t, \\ 0 & \text{otherwise,} \end{cases} \quad (7)$$

where t is a user-defined quantity used to control the deposition of

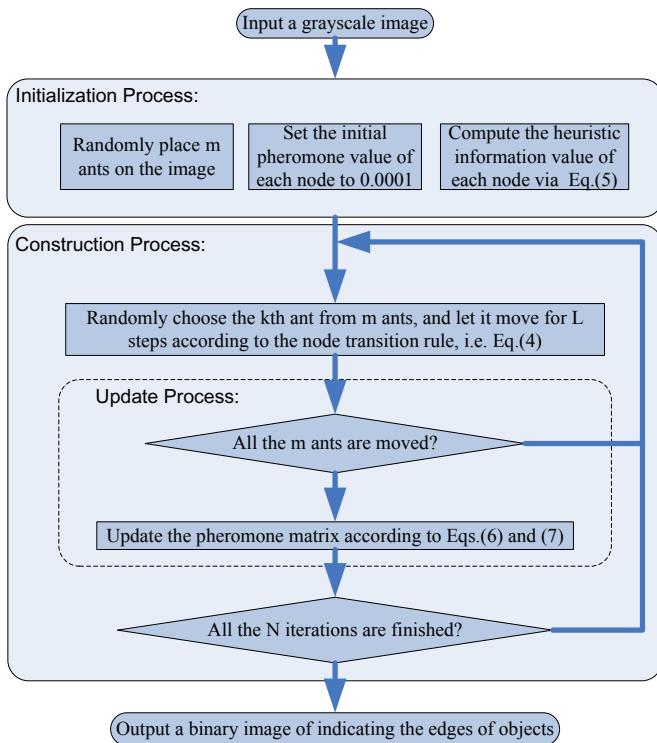


Fig. 4. Flowchart of the proposed approach.

pheromone in the pheromone update process. This parameter only allows the ants to deposit pheromone when they move following the edges with a heuristic information value above t . Furthermore, there is a daemon action that terminates the movement of the k th ant when $\eta_{(i,j)}$ is less than t or the consecutive time step is more than L . The terminated ant is immediately replaced by a new ant at the next random location. This mechanism can speed up the convergence rate of the proposed approach.

It should be noted that Eqs. (2) and (6) are different because ij and (i,j) represent edge (i,j) and node (i,j) respectively.

4) Termination criterion: once the predefined number of iterations is reached, the algorithm will stop. In this work, the number of iterations is set to 3, (i.e. construction steps $N = 3$).

After completing the above steps, the deposited pheromone information of each node could be left in image I . The higher the pheromone value, the greater is the likelihood that the node belongs to an edge. Consequently, the nodes with higher pheromone concentration will be able to represent the edges of the objects in image I .

Finally, the implementation of the proposed approach is described by the flowchart shown in Fig. 4.

4. Experimental results

The suitable parameter values of the proposed method were obtained through more than 1000 experiments. Consequently, in this section, we give the specific values of these parameters so that this method could be used conveniently. Moreover, we compare the proposed approach with two other ACO-based edge detection techniques [11,12] and four traditional edge detectors (i.e. Canny, Sobel, Log and Roberts) in the presence of noise.



4.1. Parameter setting

In the proposed method, it is difficult to determine the best parameter values because there are many parameters needing to be set. In addition, according to the characteristics of the ACO meta-heuristic approach [21], we know that the parameter setting of ACO could not be determined by mathematical derivation. Hence, in this work, a group of suitable parameter values were determined by experiments, but three of them (i.e. m , τ_{init} and N) are from the literatures [11,12,17,18]. In this section, we set the parameters in accordance with the procedure of the implementation steps of the proposed method.

1. **Parameter setting of initialization process:** the parameters m and τ_{init} are set to $\lfloor \sqrt{A} \rfloor$ and 0.0001, respectively. For a $C \times R$ image I , its area $A = C \times R$, and $\lfloor \cdot \rfloor$ represents rounding down to integer (i.e. the largest integer that does not exceed its argument). The two parameters are used in the previous ACO-based edge detection methods [11,12,17,18], thus, we also directly adopted in the proposed method.
2. **Parameter setting of construction process:** there are four parameters needing to be set in this process, namely, the time steps of ant moving (L), the memory length of ant (l), the control factor of the pheromone intensity (α) and the control factor of the heuristic information (β). Parameters L and l affect the performance of the proposed method, dependent on the image size, because L and l may influence the ant moving distance and the communication between ants, respectively. If the image size is large, L and l are small, then leading to fewer edges detected. If the image size is small, L and l are large, then leading to more edges detected but more running time. Hence, a trade-off between the two parameters and the image size is necessary. According to experimental observations and the two parameter values used in the previous methods [11–18], L and l are set to $\lfloor 3\sqrt{A} \rfloor$ and $\lfloor \sqrt{P} \rfloor$, respectively. For a $C \times R$ image I , its perimeter $P = 2 \times (C + R)$. The influences of L and l on the experimental results of Cameraman image with 256×256 pixels, are illustrated in Fig. 5. Parameters α and β determine the relative weight of the pheromone trail and the heuristic information, respectively. To select the two parameters, we used a synthetic square image to



Fig. 5. Experimental results of Cameraman image with 256×256 pixels. (a) $L = 256$, $l = 20$; (b) $L = 768$, $l = 30$; and (c) $L = 1024$, $l = 40$.

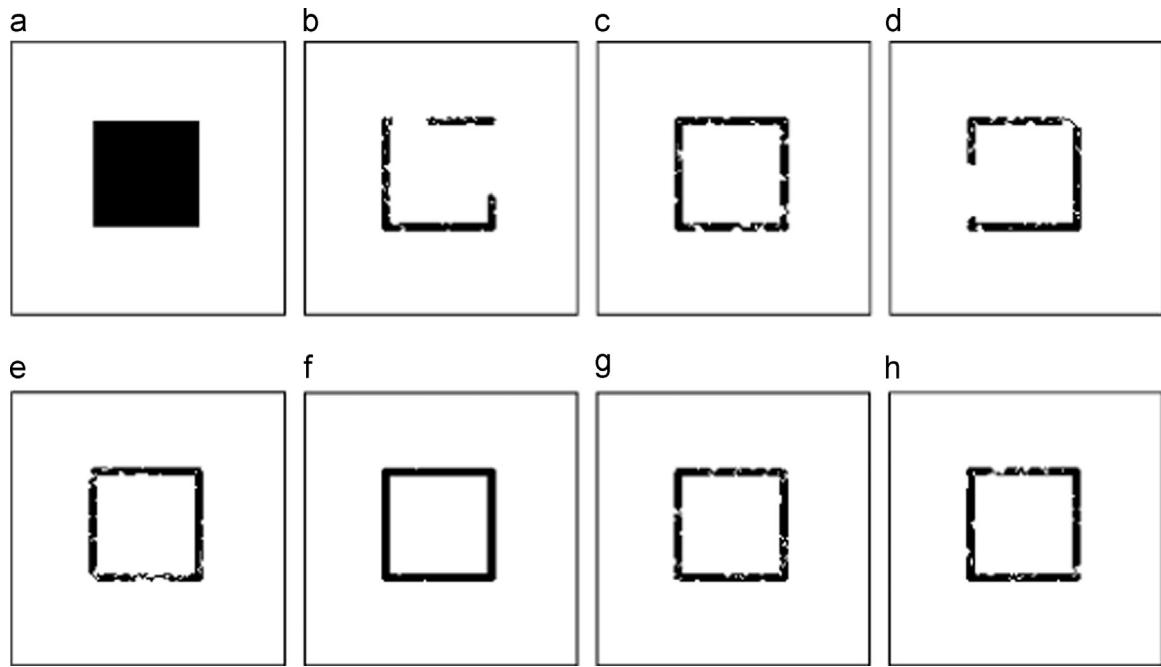


Fig. 6. Experimental results of a synthetic square image with 128x128 pixels. (a) Synthetic image; (Top) $\alpha = 1$; (b) $\beta = 1$, (c) $\beta = 2$, (d) $\beta = 3$; (Bottom) $\beta = 2$; (e) $\alpha = 1.5$, (f) $\alpha = 2$, (g) $\alpha = 2.5$, and (h) $\alpha = 3$.

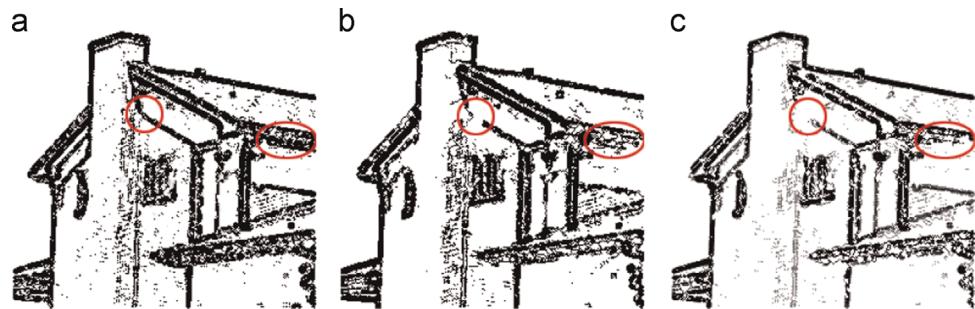


Fig. 7. Experimental results of House image, 256 × 256 pixels, using different values of ρ . (a) $\rho = 0.02$; (b) $\rho = 0.04$; and (c) $\rho = 0.06$.

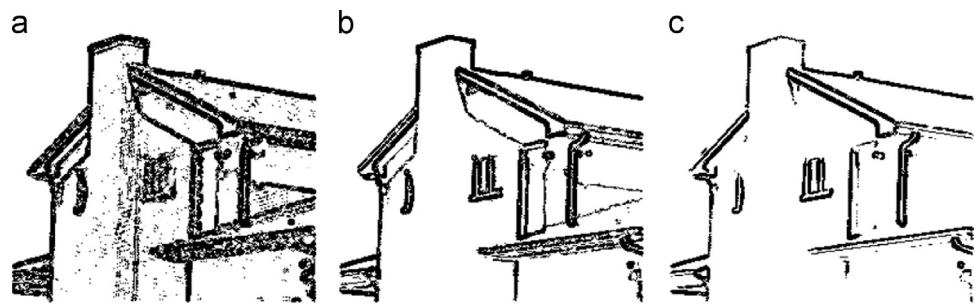


Fig. 8. Experimental results of House image, 256 × 256 pixels, using different values of t . (a) $t = 0.08$; (b) $t = 0.1$; and (c) $t = 0.12$.

- illustrate the influences of them, some experimental results are shown in Fig. 6. After many trials and errors, the two parameters are set to $\alpha = 2$, $\beta = 2$, yielding reasonably good results.
3. Parameter setting of update process: parameters ρ and t control the pheromone update. The higher the value of ρ , the faster the pheromone evaporates, and the less details of the edge will be detected, as shown in Fig. 7. In addition, Fig. 8 shows the role of t , the larger the value of t , the less noise will be detected. To further discuss the influences of t on noise type and noise level, we carried out more experiments on House image with Gaussian noises (mean value is 0, variance values are 0.01 and 0.03, shown in the first two rows of Fig. 9.) and Salt and Pepper noises (noisy densities are 0.02 and 0.04, shown in the latter rows of Fig. 9.).
 - The experimental results in Fig. 9 show that the proposed method could suppress Gaussian noise better than Salt and Pepper noise, and with the increase of the noise level, the proposed method will gradually lose its effectiveness. According to these experimental results, here, ρ and t are chosen as 0.02 and 0.1, respectively.
 4. Parameter setting of decision process: parameter N is used as the termination criterion, its value can be chosen from 2, 3 and 4, on the basis of the literatures [11–18]. By experimental

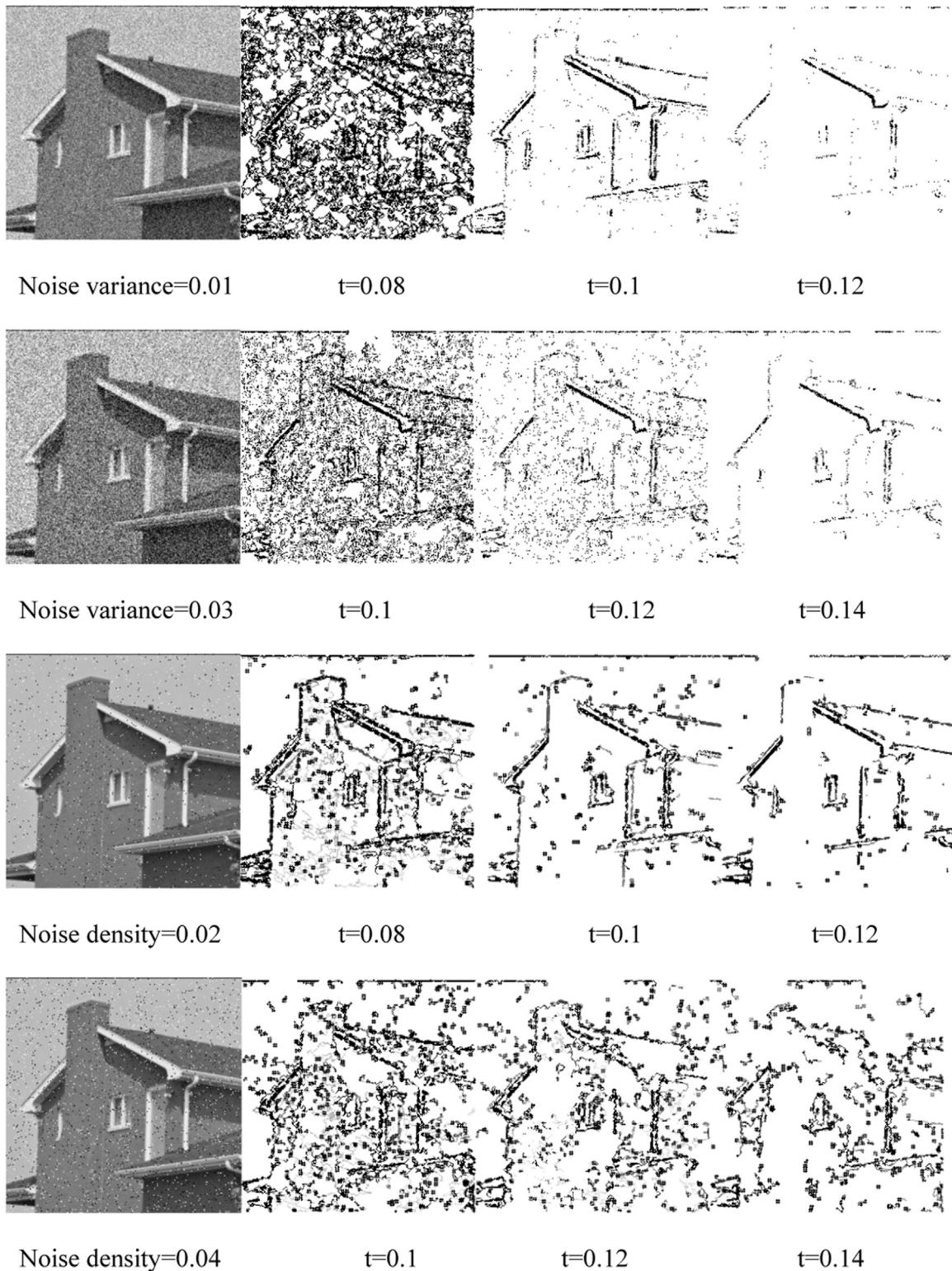


Fig. 9. Experimental results of House noisy images, 256 × 256 pixels, using different values of t .

observations, $N = 3$ could give reasonably good results, as shown in Fig. 10.

Finally, all the determined parameter values are listed in Table 1. In order to further verify the effectiveness of these parameter values, we carried out experiments on 50 real images (30 images are from the University of Southern California – Signal and Image Processing Institute (USC-SIPI) image database, other 20

images are from our laboratory). Fig. 11 shows five experimental results from these experiments. Although the sizes of the five images are different, the object edges in these images can be detected effectively.

Note that these parameters are determined by trial and error, and hence in the future we will further study the reasonable derivation of the parameter setting.

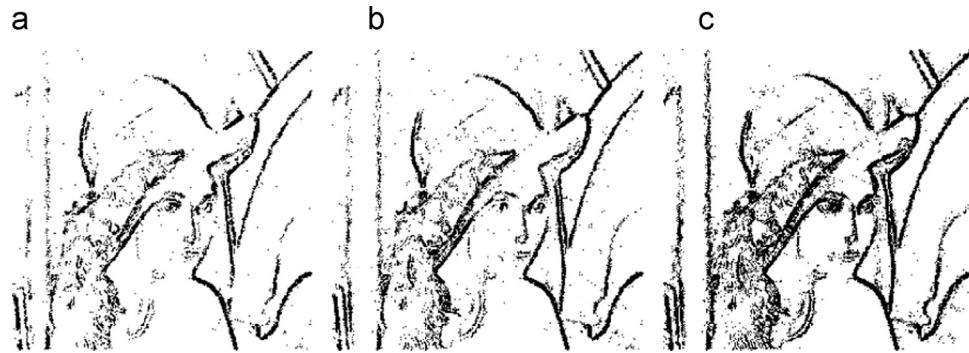


Fig. 10. Experimental results of Lena noisy image with Gaussian noise (mean value is 0, variance value is 0.01). 256 × 256 pixels, using different numbers of iterations. (a) $N = 2$; (b) $N = 3$; and (c) $N = 4$.

Table 1
Suitable parameter values of the proposed approach

Symbols	η_{init}	m	L	l	α	β	ρ	t	N
Values	0.0001	$\lfloor \sqrt{A} \rfloor$	$\lfloor 3\sqrt{A} \rfloor$	$\lfloor \sqrt{P} \rfloor$	2	2	0.02	0.1	3

4.2. Performance comparison

Because noise is a common element present in images, in this section, the proposed method will be compared with other two ACO-based edge detection techniques [11,12] and four conventional edge detectors (i.e. Canny, Sobel, Log and Roberts) in the presence of noise.

4.2.1. Compared with other ACO-based edge detection methods

The other two ACO-based edge detection methods were proposed by Nezamabadi-pour et al. [11] and Tian et al. [12], respectively. Since Nezamabadi-pour et al. applied AS algorithm and Tian et al. employed ACS algorithm, the two methods are representative of ACO-based edge detection techniques. Although the edge thinning operation was employed in the Nezamabadi-pour's method, to be fair, we just compared the edge detection results directly from ACO algorithms and neglected the thinning

process, because this process is often referred to as a post-processing procedure. In addition, we used the parameters recommended in each method to carry out experiments and make the comparisons as fair as possible.

Six standard test images (i.e. Cameraman, House, Pepper, Lena, Airplane and Pine) with Gaussian noise (mean value is 0, variance value is 0.01) were employed in the comparison experiments. The qualitative experimental results are shown in Fig. 12. In the visual comparison, the edges detected by the proposed method are more complete and robust.

Even though the proposed approach visually outperforms the other two methods, it is also necessary to quantitatively evaluate the performances of the three methods. Due to without edge thinning operation, here, we only adopted three evaluation indicators (i.e. completeness, discriminability and robustness) in [22]. The three indicators measure the abilities of an edge detector to mark all possible edges, to discriminate between important and not important edges and to suppress noise, respectively. In addition, we compared the running time of the three methods implemented by MATLAB programming language on a PC computer with Pentium4 CPU 2.4 GHz and 1 GB RAM. These quantitative comparison results of the three methods are shown in Fig. 13.

It can be seen from Fig. 13(a)–(c) that the completeness, the



Fig. 11. Experimental results of five different sizes of images. (Top) Test images: the sizes of them from left to right are 256 × 256, 360 × 270, 512 × 512, 640 × 480 and 1280 × 960 pixels. (Bottom) Detected edges of these test images.



Fig. 12. Experimental results of the three methods on Cameraman, House, Peppers and Lena noisy images, 256 × 256 pixels. From left to right, (a) Noisy images; (b) Nezamabadi-pour et al.'s method; (c) Tian et al.'s method; and (d) proposed method.

discriminability and the robustness of the proposed method outperform the other two ACO-based edge detection methods. The main reasons are as follows:

First, the three methods adopt different structures to compute the heuristic information function. The region sizes of three

structures shown in Fig. 2 are 8, 16 and 24, respectively. The difference of intensity value between two nodes with the same color in the three structures is similar to the intensity gradient. Thus, the structure adopted in the proposed method could compute higher gradient values than two other methods. In other words, the

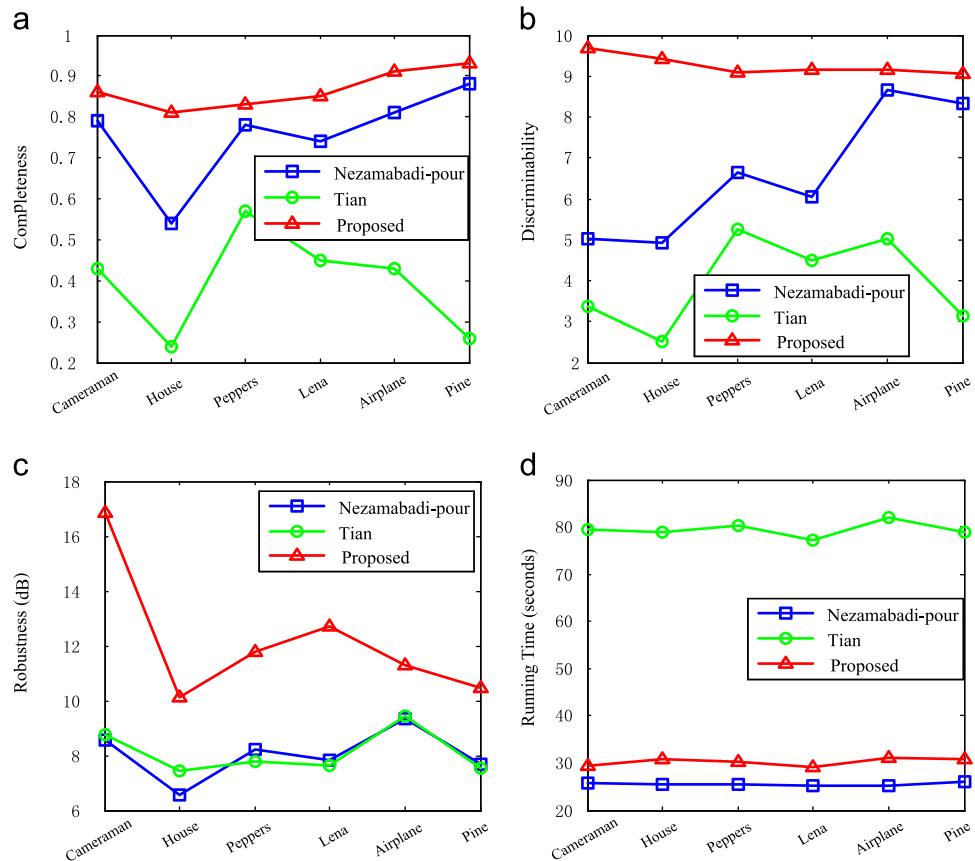


Fig. 13. Quantitative comparison results of the three methods on six noisy images. (a) completeness; (b) discriminability; (c) robustness; and (d) running time.

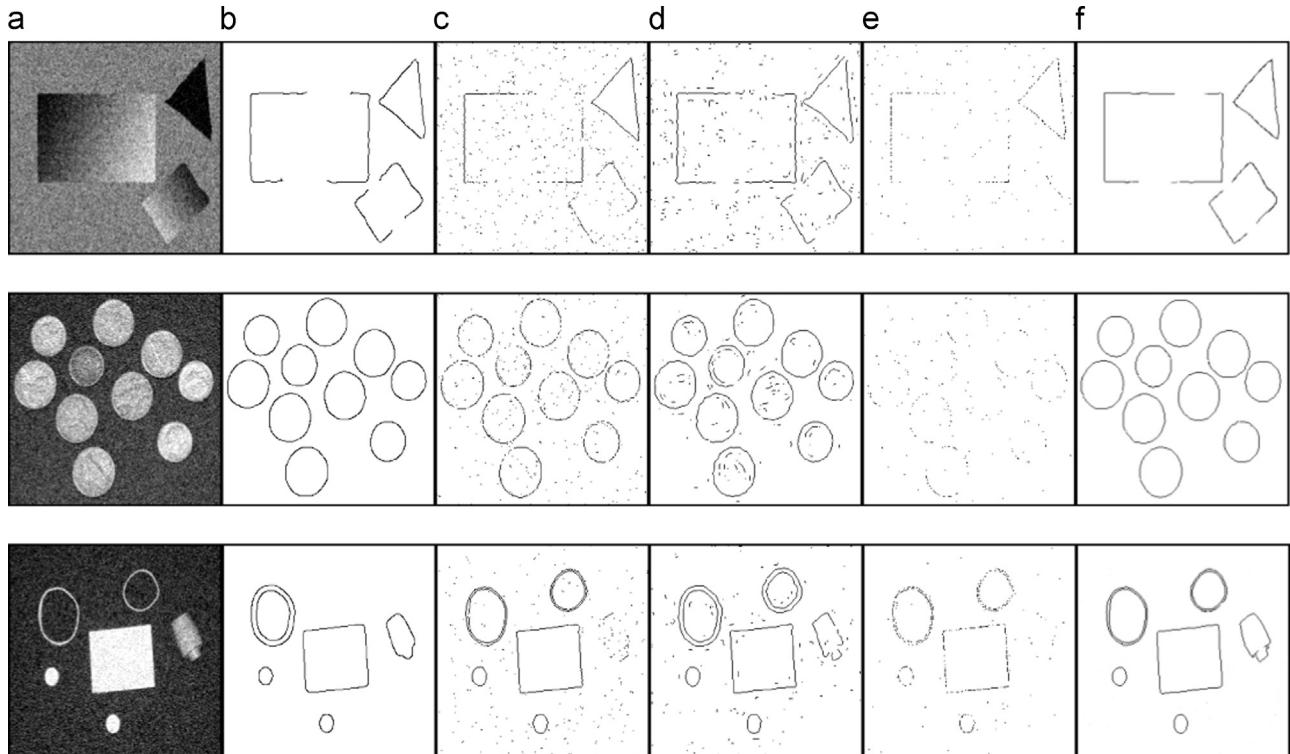


Fig. 14. Experimental results of the five methods on three noisy images, 256 × 256 pixels. (a) Noisy image; (b) Canny; (c) Sobel; (d) Log; (e) Roberts; and (f) proposed method.

Table 2

Quantitative comparison results of the five methods on the synthetic image.

	Canny	Sobel	Log	Roberts	Proposed
Completeness	0.63	0.70	0.50	0.20	0.87
Robustness	26.84	23.04	21.30	23.84	27.28
Accuracy	0.65	0.76	0.34	0.55	0.89
Running time (s)	0.45	0.64	0.19	0.11	23.20

Table 3

Quantitative comparison results of the five methods on Coins image.

	Canny	Sobel	Log	Roberts	Proposed
Completeness	0.55	0.71	0.54	0.21	0.88
Robustness	26.36	23.24	21.32	20.43	26.49
Accuracy	0.54	0.79	0.51	0.58	0.85
Running time (s)	0.30	0.32	0.22	0.16	25.14

Table 4

Quantitative comparison results of the five methods on Pillsetc image.

	Canny	Sobel	Log	Roberts	Proposed
Completeness	0.22	0.82	0.64	0.20	0.90
Robustness	26.96	23.05	21.52	19.52	27.11
Accuracy	0.31	0.68	0.53	0.64	0.86
Running time (s)	0.73	0.23	0.30	0.11	24.26

gradient response on the edge in the this structure would be higher than two other methods. As a result, this structure could detect more edge points and improve the completeness and the discriminability.

Second, the other two methods used the global thresholding techniques (Nezamabadi-Pour et al. chose the mean value of the final pheromone intensity values as the global threshold value, Tian et al. applied the well-known Otsu's method to determine the global threshold value.), but the proposed method adopted a user-defined threshold value to control the deposition of the pheromone in the pheromone update process. In other words, this thresholding technique could timely suppress noise in the pheromone update process and improve the discriminability and robustness of the proposed method.

Third, according to the discussions about the parameter setting in Section 4.1, we know that the selection of the parameters could also affect the performances of the ACO-based edge detection techniques. For example, parameters ρ and t play important roles in detecting the details of image edge and suppressing noise.

It can be observed from Fig. 13(d) that the running time of Nezamabadi-pour's method is less than Tian's method and the proposed method. The reasons would be that Nezamabadi-pour's method spends less time on computing the heuristic information function, and Tian's method based on the ACS algorithm needs two-time pheromone updating operation (see Section 2.2), thus, the running time of Tian's method is longer.

4.2.2. Compared with traditional edge detectors

We compared the proposed method with traditional edge detectors such as Canny, Sobel, Log and Roberts on one synthetic image and two real images (i.e. Coins and Pillsetc) with Gaussian noise (mean value is 0 and variance is 0.02). Since the edges detected in the proposed method are thicker, the edge thinning operation in [23] was used as a post-processing procedure in these comparison experiments. In addition, Pratt's figure of merit in [24] as an accuracy indicator was used to measure the abilities of these edge detectors to mark edges as close as possible to ideal edges.

The qualitative experimental results are shown in Fig. 14. It is

observed that the robustness of Canny and the proposed method are higher than the other three methods, but the completeness and the accuracy of Canny are lower. For an example, one object in Pillsetc image was not detected by Canny operator, as shown in the last row of Fig. 14. It should be noted that Canny's parameters, Log and our parameters were adjusted to suppress noises effectively, parameters σ and $thres$ in Canny's method were set to 3 and 0.5, respectively, while parameters ρ and t in the proposed method were set to 0.006 and 0.13, respectively. The parameters of other three methods are default values of `edge` function in MATLAB, except that parameter σ in Log operator was set to 3.

The quantitative comparison results of the five methods on the three images are tabulated in Tables 2–4

It can be seen from Tables 2–4 that three indicators (i.e. completeness, robustness and accuracy.) of the proposed method are better than the other four traditional edge detection methods. In practice, the three indicators are critical indicators for most machine vision applications, therefore, we mainly compared the three indicators. The four conventional edge detectors were implemented by MATLAB toolbox on the same hardware platform as the previous experiments. The running time of the proposed method is the longest.

5. Conclusions

In this paper an ACO-based edge detection approach was proposed. We employed a new heuristic function to improve the gradient response on the edge and adopted a user-defined threshold in the pheromone update process to suppress noise. According to the results from literatures [11–18] and our experimental results, for ease of use, we provided a group of suitable parameter values and discussed the roles of some core parameters in detail. Such as parameters ρ and t also play important roles in detecting the details of image edge and suppressing noise. Experimental results indicated the effectiveness of the proposed approach. In the presence of noise, compared with other two ACO-based methods [11,12] and four traditional edge detection methods (i.e. Canny, Sobel, Log and Roberts), the proposed technique has better performance, but its running speed is slower than Nezamabadi-pour's method and the four conventional methods. Hence, we will further study how to improve the running speed of the proposed method.

Acknowledgments

The authors wish to thank Jing Tian, who is the author of the literature [12], for his help in the implement of the algorithms used in this work.

References

- [1] O. Laligant, F. Truchetet, J. Miteran, Merging system for multiscale edge detection, *Opt. Eng.* 44 (2005) 035602 1–11.
- [2] Q. Sun, Y. Hou, Q. Tan, C. Li, Shaft diameter measurement using a digital image, *Opt. Lasers Eng.* 55 (2014) 183–188.
- [3] Q. Sun, Y. Hou, Q. Tan, C. Li, M. Liu, A robust edge detection method with sub-pixel accuracy, *Optik* 125 (2014) 3449–3453.
- [4] L. Fan, F. Song, S. Jutamulia, Edge detection with large depth of focus using differential Haar-Gaussian wavelet transform, *Opt. Commun.* 270 (2007) 169–175.
- [5] D. Marr, E. Hildreth, Theory of edge-detection, *Proc. R. Soc. Ser. B – Biol. Sci.* 207 (1980) 187–217.
- [6] J. Canny, A computational approach to edge-detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (1986) 679–698.
- [7] P.H. Qiu, Jump surface estimation, edge detection, and image restoration, *J. Am. Stat. Assoc.* 102 (2007) 745–756.

- [8] M. Dorigo, M. Birattari, T. Stutzle, *Ant colony optimization*, *IEEE Comput. Intell. Mag.* 1 (2006) 28–39.
- [9] X. Zhuang, Edge feature extraction in digital images with the ant colony system, in: IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, vol. 197, 2004, pp. 133–136.
- [10] X. Zhuang, Image feature extraction with the perceptual graph based on the ant colony system, in: IEEE International Conference on Systems, Man & Cybernetics, vol. 7, no. 1, 2004, pp. 6354–6359.
- [11] H. Nezamabadi-pour, S. Saryazdi, E. Rashedi, *Edge detection using ant algorithms*, *Soft Comput.* 10 (2006) 623–628.
- [12] J. Tian, W. Yu, S. Xie, *An ant colony optimization algorithm for image edge detection*, *IEEE Congr. Evol. Comput.* 8 (1) (2008) 751–756.
- [13] A. Jevtic, J.D. Quintanilla, G.M.J. Cortina, D. Andina, Edge detection using ant colony search algorithm and multiscale contrast enhancement, in: Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics (Smc 2009), vol. 1–9, 2009, pp. 2193–2198.
- [14] A.V. Baterina, C. Oppus, Ant colony optimization for image edge detection, in: Proceedings of the 9th Wseas International Conference on Signal Processing Robotics and Automation, Ispra '09, vol. 342, 2010, pp. 220–225.
- [15] A. Jevtić, B. Li, Ant algorithms for adaptive edge detection, in: Taufik Abrão (ed.), *Search Algorithms for Engineering Optimization*, 2012, pp. 953–978.
- [16] K. Benhamza, H. Merabti, H. Seridi, Adaptive edge detection using ant colony, in: 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA), 2013, pp. 197–202.
- [17] C. Gupta, S. Gupta, *Edge detection of an image based on ant colony optimization technique*, *Int. J. Sci. Res.* 2 (2013) 114–120.
- [18] S. Ari, D.K. Ghosh, P.K. Mohanty, Edge detection using ACO and F ratio, *Signal Image Video Process.* 8 (2014) 625–634.
- [19] D.S. Lu, C.C. Chen, *Edge detection improvement by ant colony optimization*, *Pattern Recognit. Lett.* 29 (2008) 416–425.
- [20] A. Jevtic, I. Melgar, D. Andina, Ant based edge linking algorithm, in: Proceedings of the 35th Annual Conference of IEEE Industrial Electronics, lecon, vol. 6, no. 1, 2009, pp. 3177–3182.
- [21] M. Dorigo, G.D. Caro, L.M. Gambardella, *Ant algorithms for discrete optimization*, *Artif. Life* 5 (1999) 137–172.
- [22] R. Moreno, D. Puig, C. Julia, M.A. Garcia, A new methodology for evaluation of edge detectors, in: Proceedings of the 16th IEEE International Conference on Image Processing (ICIP), 2009, pp. 2157–2160.
- [23] E.R. Davies, *Computer and Machine Vision: Theory, Algorithms, Practicalities*, fourth ed., Academic Press, Kidlington (2012), p. 251–254.
- [24] I.E. Abdou, W.K. Pratt, *Quantitative design and evaluation of enhancement/thresholding edge detectors*, *Proc. IEEE* 67 (1979) 753–763.