Expert Systems with Applications 40 (2013) 4315–4332

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Ant algorithms for image feature extraction

Rob J. Mullen, Dorothy N. Monekosso, Paolo Remagnino [*]

Robot Vision Team, Kingston University, Penrhyn Road Campus, KT1 2EE, Surrey, United Kingdom

ARTICLE INFO

Keywords:
Ant algorithm
Swarm intelligence
Feature extraction
Image processing

ABSTRACT

This paper extends on previous work in applying an ant algorithm to image feature extraction, focusing on edge pattern extraction, as well as the broader study of self-organisation mechanisms in digital image environments. A novel method of distributed adaptive thresholding is introduced to the ant algorithm, which enables automated distributed adaptive thresholding across the swarm. This technique is shown to increase performance of the algorithm, and furthermore, eliminates the requirement for a user set threshold, allowing the algorithm to autonomously adapt an appropriate threshold for a given image, or data set. Additionally this approach is extended to allow for simultaneous multiple-swarm multiple-feature extraction, as well as dynamic adaptation to changing imagery.

1. Introduction

This paper reports research into the application of swarm intelligence to image feature extraction, and specifically, the use of an ant-algorithm approach based on the original Ant System (AS) (Dorigo & Gambardella, 1997). Analysis of the algorithm is carried out on two levels: with respect to the use of ant-algorithms for facilitating emergent self-organisation and pattern formation at the swarm level, in response to local agent–agent and agent–environment interactions, by imposing simple decision making rules on simple artificial agents; and with respect to the application of image feature extraction. Initial findings of the proposed algorithm were reported in Mullen, Monekosso, Barman, Remagnino, and Wilkin (2008). This paper extends on the previous work with additional experiments, and significant developments on the proposed approach, namely a distributed dynamic threshold technique; as well as the application to dynamic imagery for image feature tracking.

The self-organisation process studied in this paper can be likened to the use of *templates*, where a template is a pattern used to construct another pattern (Bonabeau, Dorigo, & Theraulaz, 1999). An example in nature is where an ant colony builds a pattern of walls around a brood pile in the environment, which becomes the nest. Similar environment-based templates are used to stimulate self-organised pattern formation in this paper.

The distributed nature of swarm intelligence methods such as the ant-algorithm presented in this paper allow us to further utilise such nature inspired approaches in order to further increase efficiency in the self-organising process and consequently improve the problem solving performance. Inspired by the distributed and adaptive specialisation observed in social insects this paper explores parameter adaptation, exploiting the distributed nature of the embedded swarm by allowing individual agents within the swarm to adapt their individual parameters for a given environment.

A number of studies suggests that individual experience shapes behavioural ontogeny, and that thresholds for responding to given stimuli may be dynamic, rather than static (see for example Calderone & Page, 1996 and a short survey can be found in Bonabeau et al. (1999)). In this paper, focus is placed on adaptive specialisation through dynamic thresholds, giving individual agents the ability to change their behaviours in response to perceived changes in the environment.

Additionally, this paper introduces dynamic environments such that the swarm has the additional challenge of adapting to changing environment structures, involving adaptive self-organisation and pattern formation.

2. Related work

Digital image processing is well established amongst the scientific community and there exists many methods for performing various image processing tasks. Although certain aspects of machine learning and artificial intelligence have been utilised in image processing for some time, the use of ant algorithms to perform image processing tasks is a relatively new technique. Ant algorithms have been used for basic low level image segmentation via boundary detection methods (Fernandes, Ramos, & Rosa, 2005b; Nezamabadi-pour, Saryazdi, & Rashedi, 2006; Ramos & Almeida, 2000) and via clustering methods (Channa, Rajpoot, & Rajpoot, 2006; Ouadfel & Batouche, 2002). In Ma, Tian, and Yu (2010)

[*] Corresponding author. Tel.: +44 (0) 20 8417 2930; fax: +44 (0) 208417 2972.
E-mail address: P.Remagnino@kingston.ac.uk (P. Remagnino).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.01.020

the authors present an ant-algorithm for visual saliency detection in images. The major differences between these approaches is the nature of the heuristic information used to guide the ant agents, however there is no clear advantage over one particular method. Although these works show promising results in terms of image segmentation, the respective analysis is mainly limited to that of a qualitative nature.

In Chialvo and Millonas (1995) and Rauch, Millonas, and Chialvo (1995) the authors investigate how swarms of insect-like agents form patterns and build cognitive maps by creating a network of trails. The authors in Ramos and Almeida (2000) follow a similar approach to investigate the self-organising nature of a swarm of artificial ants in response to image edge features in a digital image habitat. These works introduce more quantitative analysis of the self-organising properties of the algorithms (but not the application of image processing).

Extensive work has been carried out to study self-organisation in swarm-based systems, and in many cases this is synonymous with adaptation in terms of swarm behaviours, as self-organisation often implies some form of adaptation, for example in terms of the changing, or emerging structure of the swarm in pattern formation. An example of such adaptation in nature is the way in which bees use dancing as a means of recruitment in foraging tasks. In order to focus foraging towards the best food sources, as well as refocus foraging in response to variations in available forage, honeybees adjust both the duration and vigor of their dancing as a function of profitability of their current source (Rozenberg et al., 1999). A forager uses an internal gauge to assess the profitability of their source. The bee's nervous system has a threshold calibrated into it, which the bee uses to weigh variables when deciding whether a patch is worth foraging, and if so, whether it is worth advertising to fellow workers (Rozenberg et al., 1999). There is an analogy to be drawn here with the application of the ant swarm to digital imagery. The internal gauge of each individual ant is the threshold T, which governs whether or not an ant should deposit pheromone (thus advertising to fellow agents), depending on the local visibility, η, (i.e. the profitability) associated with a given move (details below). This paper examines a method of adaptation of the parameter T in response to variation in the visibility, akin to how honeybee's vary their dancing in response to variation in available forage.

There is little evidence in the literature of studies involving swarm-based feature extraction methods applied to dynamic imagery. In Ramos and Almeida (2000) the authors study the self-organising properties of a swarm embedded in a digital image environment, including the swarms ability to re-adapt as the image environment is swapped with another one. In Fernandes et al. (2005b) the authors examine the ability of the swarms to self-regulate their population size for given image environments, and show that regulating the population size can result in faster convergence times of the swarm to adapt to a particular image structure. The same authors extend their research in Fernandes, Ramos, and Rosa (2005a) to investigate the swarms ability to search for peaks and valleys in dynamic 3D landscapes represented by mathematical functions.

3. An ant algorithm for image feature extraction

The basic framework of this algorithm is based around the workings of the original AS (Dorigo & Gambardella, 1997), with a modified, application specific pheromone update rule and heuristic information.

The algorithm employs artificial ants as simple computational agents. The algorithm is initialised with N ants occupying 'random' pixels within the image, where *pixels* in the *image* are equivalent to *states* in the search *environment*. The aim of the ants is to locate and map out the boundaries within the image. This is achieved by introducing heuristic information that weighs higher the probability of an ant moving from its current location to the allowed surrounding pixels that have the greatest boundary characteristics (greatest change in image gradient for example). Each ant deposits an amount of pheromone with each move to a new pixel, where the amount deposited may also be a function of, e.g. change in image gradient, and pheromone evaporation occurs at a fixed rate per iteration. The transition rule is then a function of the heuristic information and the pheromone map.

Formally, at each time step t, each of the N ants moves a distance of 1 pixel to one of the eight surrounding pixels (it should be noted that working in the scale-space where agents are able to move varying distances is possible, but beyond the scope of research of this paper). Each ants transition from state to state is guided by two main factors: *heuristic information*, and *artificial pheromone trails*.

The heuristic information is defined here as the *visibility*, η, which is the local image gradient around the ants current pixel location, measured with respect to the direction of travel from the ants previous location. In the context of this work the term *heuristic information* is used to describe any information the agents use from the local environment, to influence their movement. At each time step each ant calculates the visibility, η, associated with each possible move to the eight surrounding pixels, such that for an ant at pixel location (x,y),

$$\eta_{x-1,y-1} = |(I(x-1,y) - I(x,y-1))| \tag{1}$$

$$\eta_{x-1,y} = |(I(x-1,y-1) - I(x-1,y+1))| \tag{2}$$

$$\eta_{x-1,y+1} = |(I(x-1,y) - I(x,y+1))| \tag{3}$$

$$\eta_{x,y-1} = |(I(x+1,y-1) - I(x-1,y-1))| \tag{4}$$

$$\eta_{x,y+1} = |(I(x-1,y+1) - I(x+1,y+1))| \tag{5}$$

$$\eta_{x+1,y-1} = |(I(x,y-1) - I(x+1,y))| \tag{6}$$

$$\eta_{x+1,y} = |(I(x+1,y-1) - I(x+1,y+1))| \tag{7}$$

$$\eta_{x+1,y+1} = |(I(x+1,y) - I(x,y+1))| \tag{8}$$

where $I(x,y)$ gives the image intensity value at pixel location (x,y), and η_{ij} is equivalent to $\eta_{\pm x \pm y}$ depending on which direction the ant moves in.

This problem specific heuristic information aims to guide the ant agents to follow along edges and high contrast boundary regions within the image.

The pheromone concentration at any given pixel is given by τ_{ij}. Pheromone deposition by the ants happens at the end of each time step, along with a constant evaporation of the entire pheromone field. These processes are governed by the following pheromone update rule:

$$\tau_{ij}(t+1) = (1-\rho)\cdot\tau_{ij}(t) + \sum_{k=1}^{N}\Delta\tau_{ij}^{k} \tag{9}$$

where $\rho \in \; (0,1]$ is the evaporation rate and $\Delta\tau_{ij}^{k}$ is the quantity of pheromone deposited at pixel location (i,j) by the kth ant and is given by:

$$\Delta\tau_{ij}^{k} = \begin{cases} \eta_{ij}/255 & \text{if ant } k \text{ in pixel } (i,j) \text{ and } \eta_{ij} > T \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where T is a user defined threshold value that can be set to only allow pheromone deposition by ants following edges or boundaries above a certain 'strength.' In addition there is also a *daemon action* implemented that terminates any ant agent with $\eta_{ij} < T$ for more than Z consecutive time steps. This terminated agent is immediately replaced by a new ant agent at a new 'random' location. This step is implemented to reduce the amount of ant agents 'lost' searching large background areas of the image and to speed up the rate of convergence of the agents onto the desired regions of the image search space.

Each ant then chooses its next pixel location by applying a probabilistic state transition rule, such that the probability of the *k*th ant moving to pixel location (*i*,*j*), at time step *t*, is given by

$$p_{ij}^{k}(t) = \begin{cases} \frac{[\tau_{ij}(t)]^{\alpha}\cdot[\eta_{ij}]^{\beta}}{\sum_{k\in allowed_k}[\tau_{i_kj_k}(t)]^{\alpha}\cdot[\eta_{i_kj_k}]^{\beta}} & \text{if } I(i,j) \in allowed_k \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

where *allowed*$_k$ is the eight pixels surrounding the *k*th ant, excluding any pixels in *tabu*$_k$, and α and β control the relative importance of the pheromone trail and visibility respectively. *tabu*$_k$ is a list containing the last *n* pixel locations visited by the *k*th ant, where $n = tabu_{max}$ gives the number of time steps into the past for which ants cannot re-visit previously visited pixel locations.

The input to the algorithm is a greyscale image, and the output is the emerged pheromone field, $\tau(t_{final})$. This is then converted to a binary image, τ_{BW}, to serve as the output feature image. Since the pheromone update rule (Eq. (10)) includes the threshold, all pixels containing pheromone are set as feature pixels when creating the binary feature map, such that,

$$\tau_{BW\,ij} = \begin{cases} 1 & \text{if } \tau_{ij}(t_{final}) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

4. A case study in leaf pattern extraction

Leaf shapes and venation patterns offer a wide ranging array of naturally occurring patterns for use in this study, and furthermore they offer an interesting and challenging image feature extraction problem. For this case study, a dataset of live leaf specimen from the *Quercus* family were collected from Royal Botanic Gardens, Kew (RBG KEW) Gardens, and captured into digital form using a high resolution flatbed scanner at RGB KEW[1].

4.1. Qualitative analysis

As the algorithm runs, the ant agents converge onto the boundary regions of the image, as illustrated in Fig. 1, which shows the ant agents positions within the image space as black pixels on a white background. The corresponding emergent pheromone field can be seen in Fig. 2, where the brighter pixels correspond to higher pheromone intensity at that pixel location within the image search space. Fig. 3 shows example results of the final pheromone field next to the corresponding ground truth image and original image for both a real and artificial leaf image. As we can see, the resultant pheromone field maps out the boundaries within the image, showing clearly the leaf outline and primary venation pattern, where large amounts of pheromone have built up.

Closer visual inspection reveals some inevitable limitations. The primary venations in Fig. 3(a) branching off the main vertical spine actually continue all the way to the edge of the leaf, however they become very thin as they get closer to the leaf edge and this makes them much less discernible even to the human eye. The effects of noise and non-uniform lighting at this level have a much greater affect, and as can be seen in Fig. 3(c), these 'weak' low contrast edges are not represented in the algorithm output. Such missing information could cause problems for leaf type classification, as closely related venation patterns might not be picked up, and unrelated patterns might be wrongly grouped together (Wilkin, 2008).

The quality of the specimen can vary significantly between samples, with some specimen exhibiting much more defined venation patterns than others, and aside from the effects of noise, as well as lighting issues inherent in the scanning process, the problem is worsened by the fact that all the images have been downsized significantly from their original scanned size.

By lowering the threshold value *T* it is possible to allow pheromone to be deposited at these lower contrast edges, however this is at the cost of increasing the effects of noise in the final pheromone field. Also, and again due to the low contrast of these edges, the pheromone trails in these regions often become disjointed, as the variation in intensity levels due to noise here is comparable to the variation due to the edges of the venation pattern.

The images shown in Figs. 1–3 are typical of the results seen in all the leaf images used in this study.

4.2. Quantitative analysis

The quantitative performance analysis of the algorithm is carried out by computing the sensitivity, specificity and accuracy of the algorithm output when compared to ground truth images. Ground truth images for the leaf images were created by manually tracing the leaf outlines and primary venation patterns via a touchscreen tablet PC device.

When analysing the performance of an image feature extraction algorithm, traditionally the results would focus on the end output of the algorithm (for example a binary pixel classification). Although the performance of the ant-algorithm approach for the specific purpose of image feature extraction is indeed of interest to this work, of particular interest is the examination of the self-organisational properties of this approach. One of the key properties to characterise when a given pheromone is self-organised is the creation of spatiotemporal structures in an initially homogeneous medium (Bonabeau et al., 1999). As such, it would be more instructive to look at the statistical results over time, throughout the self-organising process.

The results in Fig. 4 are averages from the results of twenty real leaf images. The results show how the accuracy, sensitivity and specificity vary as the algorithm runs. As can be seen, the overall accuracy increases over time, as the ant agents converge on the edges and the pheromone concentration here increases such that more and more pixels are detected as edge pixels. This results in an increase in the number of true positive classifications and a decrease in false negative, which is also reflected in the increase in sensitivity (Fig. 4(b)). The specificity (Fig. 4(c)) decreases over time as the number of true negative counts decreases and the false positive increases, as pheromone concentration builds up across the entire pheromone field, including areas outside of the 'true' edge regions. The specificity does not however decrease by any large amount and remains at a high value due to the fact that the TN count is always much greater than the other counts because the majority of the pixels within all of the images are in fact not edge or boundary pixels (i.e. they are background pixels).

Fig. 5 compares the results obtained from a real leaf image with high contrast venation pattern, to those obtained from an artificial leaf image. The accuracy, sensitivity and specificity are all higher for the artificial image (this is true in general for all the artificial leaf images). This is perhaps not surprising since the artificial images are noise free, such that the backgrounds, both inside and outside of the leaf, are of uniform grey level intensity. This means that the only change in image gradient occurs at the leaf outline edge and the venation pattern edges, so the scope for error here is minimal. There will however still be a small error count, and this is due to the fact that pixels directly next to 'true' edge pixels may

[1] Royal Botanic Gardens, Kew, is a scientific institution which carries out research in systematics, biological interactions, economic botany, conservation and horticulture (Royal Botanic Gardens, 2011). The Royal Botanic Gardens, Kew is an expansive landscape of living plants.

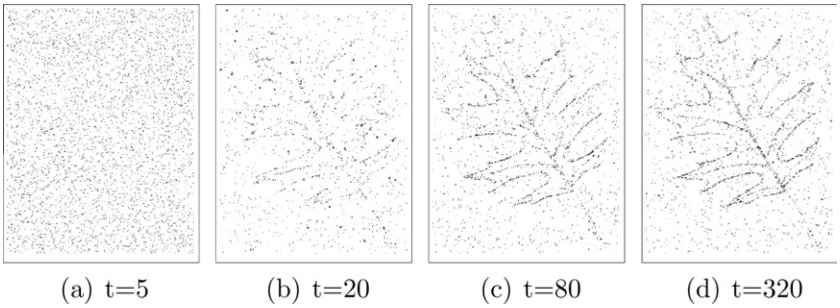


Fig. 1. Ant agent positions (black pixels) within a digital leaf image at different time-steps in the algorithm run.

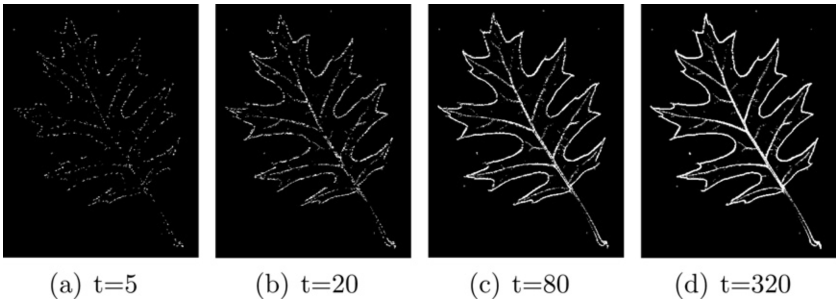


Fig. 2. Emerging pheromone map at different time-steps in the algorithm run. Brighter pixels equal higher pheromone concentration at that point (please refer to a digital copy for clear images).

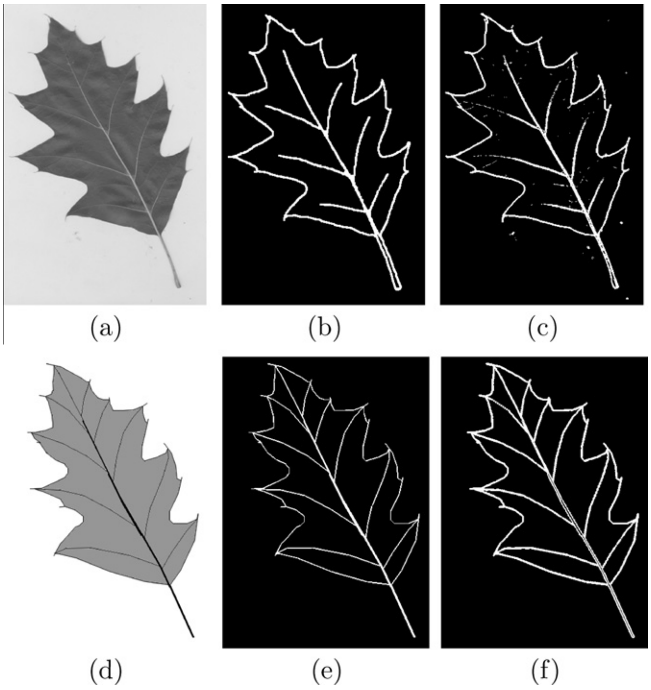


Fig. 3. Example results: original images, (a,d), corresponding ground truth images, (b,e), and final pheromone fields, (c,f). Original image (a) is a real leaf image, and original image (d) is an artificial leaf image.

also receive a large amount of pheromone, hence this is why the accuracy is not at 100 percent for the artificial image in Fig. 5(a). From an image processing perspective this might initially be thought of as a disadvantage due to the reduction in accuracy. However, there is an alternative viewpoint in that the 'blurring' effect the cumulative pheromone effect can have over the image edge features, can in some way provide a buffer for the subjective definition of where in fact the true edge feature lies.

The above analysis was carried out using an empirically determined threshold value. Fig. 6 shows a plot of the Receiver Operator Characteristic (ROC) curve of the ant-algorithm applied to a typical leaf image, with varying threshold T values. The results show a ROC curve approaching the desired $(0,1)$ point of perfect classification.

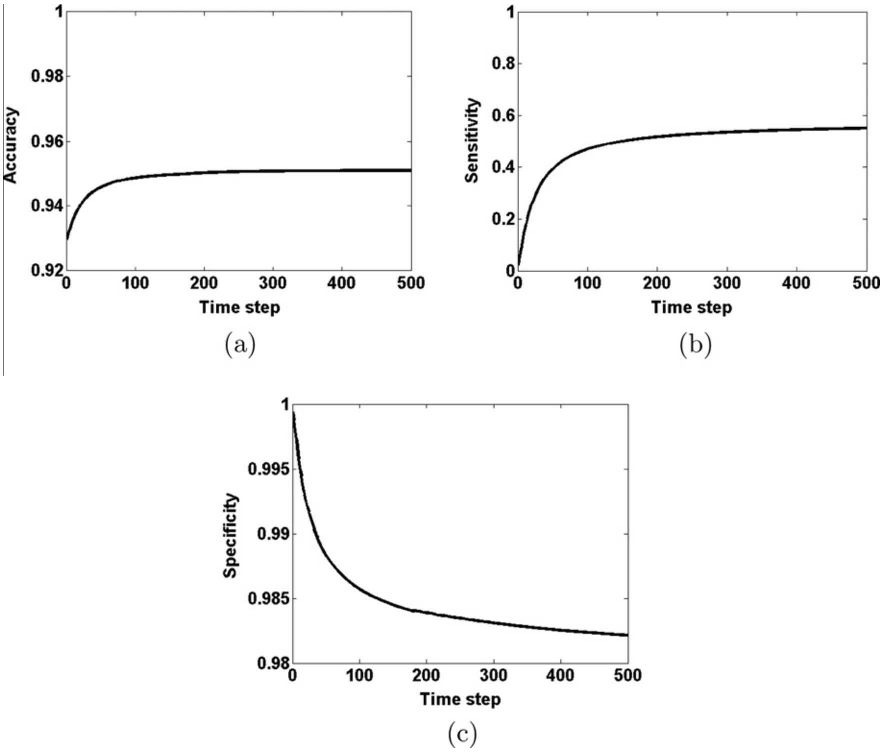
4.3. Comparison to existing methods

Here, the proposed algorithm is compared with existing methods of image edge feature extraction. The Sobel and Canny edge detector algorithms are applied separately to the same test dataset of real leaf images and the average accuracy is compared to that obtained with the ant-algorithm approach. This comparison is included to show how the ant-algorithm performs in comparison to existing methods for the specific problem of image edge feature extraction. Image processing application aside, since the goal of the ants is to self organise in response to the image edge features (thus providing the pheromone map for feature extraction), this comparison does give an indication of the extent to which the swarm has managed to self organise, by comparing to two well established methods for dealing with image edge feature extraction.

From Table 1 we see a comparable performance between the three algorithms when comparing the average accuracy (taken as an average over the leaf image dataset described previously).

Fig. 7 shows a plot of *sensitivity* versus 1 − *specificity*, showing results in ROC space of the ant-algorithm, Canny and Sobel edge detectors, applied to the dataset of 20 leaf images from RBG KEW. We see the ant-algorithm results clustering closer to the $(0,1)$ point of perfect classification, with the Canny and Sobel performing comparatively in this test.

Looking at Fig. 8 we can make a qualitative comparison between the three methods. From visual inspection we notice how the Canny method has picked up a greater amount of detail in the image gradients, which can account for the slightly lower aver-

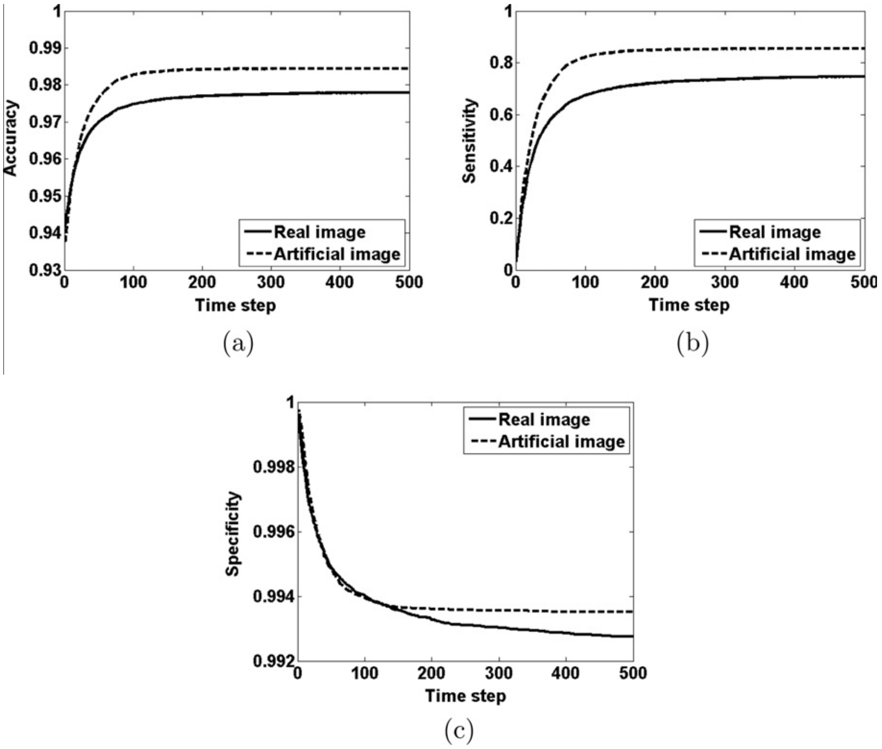


(a)

(b)

(c)

Fig. 4. Plots of average results from 20 real leaf images. Plots show average accuracy, (a), sensitivity, (b), and specificity, (c), over 500 time-steps, as measured from comparing the algorithm output with the ground truth images. Note: Scales are not the same for each image.



(a)

(b)

(c)

Fig. 5. Plots comparing results from an artificial leaf image and from a real leaf image with high contrast venation pattern. Plots show accuracy, (a), sensitivity, (b), and specificity, (c), over 500 time-steps, as measured from comparing the algorithm output with the ground truth images. Note: Scales are not the same for each plot.

age accuracy (due to over segmentation with regards to the ground truth data). One positive feature of the Canny method is the low level of discontinuities along the major contours. To an extent the same can be said for the Sobel method, when comparing to the ant algorithm method, which suffers more from discontinuities along the contours. The Canny and Sobel methods (being edge detectors) both show edge responses on each side of the larger venations. The ant algorithm on the other hand shows larger areas

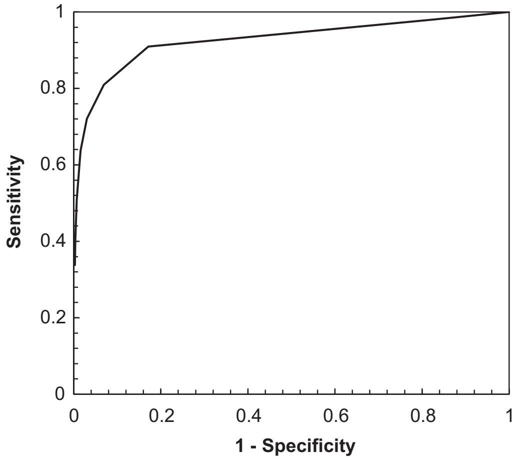


Fig. 6. Receiver Operator Characteristic (ROC) plots showing the ROC curve for the ant-algorithm with varying threshold *T*.

Table 1
A comparison of the average accuracy obtained using the proposed ant algorithm approach, the Sobel edge detector and the Canny edge detector. Experiments are carried out on the leaf data-set, as described in Section 4.2.

Algorithm	Average accuracy
Ant	0.945
Sobel	0.940
Canny	0.897

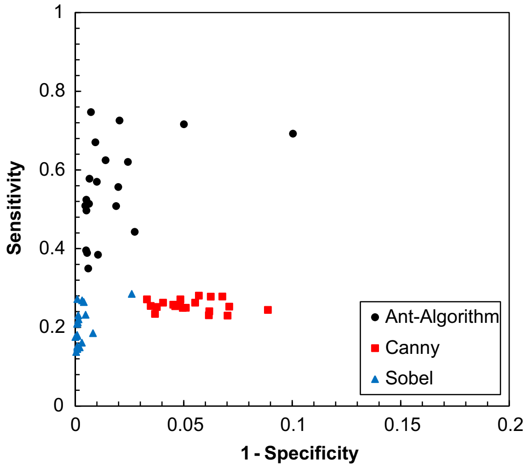


Fig. 7. Receiver Operator Characteristic (ROC) plot showing a comparison in ROC space of the results from the ant-algorithm, Canny, and Sobel edge detectors applied to the dataset of 20 leaf images from RBG KEW. Note the different scales of the *x* and *y* axis.

of detection, covering whole venations, as opposed to just the edges. This is due to the reinforcement nature of the pheromone map, whereby a build-up of pheromone occurs on and around these strong image gradients, representing the venation structure and leaf outline. This offers greater flexibility for feature extraction.

5. Multiple swarms for multiple features

If we again turn our attention to nature, there are numerous examples of ant colonies being made up of different types of specialist ants to carry out different specific tasks (Arcaute et al., 2009; Chittka & Muller, 2009; Reznikova & Novgorodova, 1998). In this same way, the proposed algorithm can be modified to employ multiple swarms to simultaneously search the image environment, with each swarm programmed differently (for example with different parameter settings, or alternative heuristic information).

An issue identified in the earlier case-study on venation feature extraction was the difficulty in discerning the smaller, tertiary venations in the extraction process. This problem was largely due to limiting factors in image quality and resolution. When setting the threshold *T* low enough to extract tertiary venation, this was detrimental to the extraction of the primary venation pattern due to the resulting evolved pheromone map including many broken segments where the tertiary venation was not well defined, and where background noise in the image had been extracted as features.

By employing two swarms, each using a different threshold, it is possible to allow them to simultaneously self-organise and evolve two separate pheromone maps; one for the primary venation pattern and leaf outline, and one for tertiary patterns[2]. Fig. 9 gives two examples using leaves from the RBG KEW dataset. For these examples we have two swarms, with $N = 3000$ for each. Swarm *A*, which has a pheromone map depicted in red[3](Fig. 9(c) and (g)), has a threshold of $T = 25$. Swarm *B* has a dual threshold of $T_{upper} = 25$,$T_{lower} = 15$, with a pheromone map depicted in green (Fig. 9(d) and (h)). The upper and lower thresholds limit Swarm *B* to edge features within a given range. The combined pheromone maps (Fig. 9(b) and (f)) show the leaf outline, primary and tertiary venation pattern, as extracted by the two swarms.

From a swarm intelligence perspective this is an interesting visual example of simultaneous self-organisation of two specialist swarms acting independently from one another. In employing this nature inspired technique it has been demonstrated visually how a simple extension to the implementation of the ant-algorithm to allow multiple swarms to run independently has facilitated dual feature extraction from the digital image environments, and furthermore, has enabled an alternative solution to a typical image processing problem.

5.1. RGB Feature extraction

Thus far only image edge gradient has been considered as the heuristic information to guide the ant agents. It would be interesting to study the effects on the self-organisation and pattern formation using different heuristic information, to guide the agents towards alternative image features.

The images used thus far have been greyscale. Using RGB colour images, the heuristic information can be changed to guide the ants towards areas of the image with a prominent intensity of a particular colour.

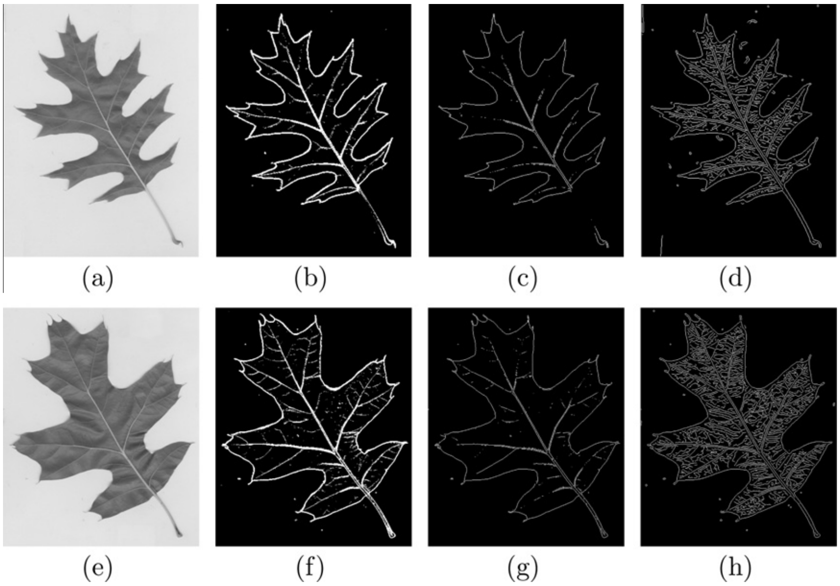
For example, if the heuristic information is defined as,

$$\eta_{ij} = (R_{ij} - B_{ij}) + (R_{ij} - G_{ij}) \tag{13}$$

where *R*, *G* and *B* are the *Red*, *Green* and *Blue* channels of the image respectively, then the visibility for the ants becomes a weighted average of the red channel in the RGB image, such that the ants will converge on regions of the image with a prominent red intensity, with $T = 30$.

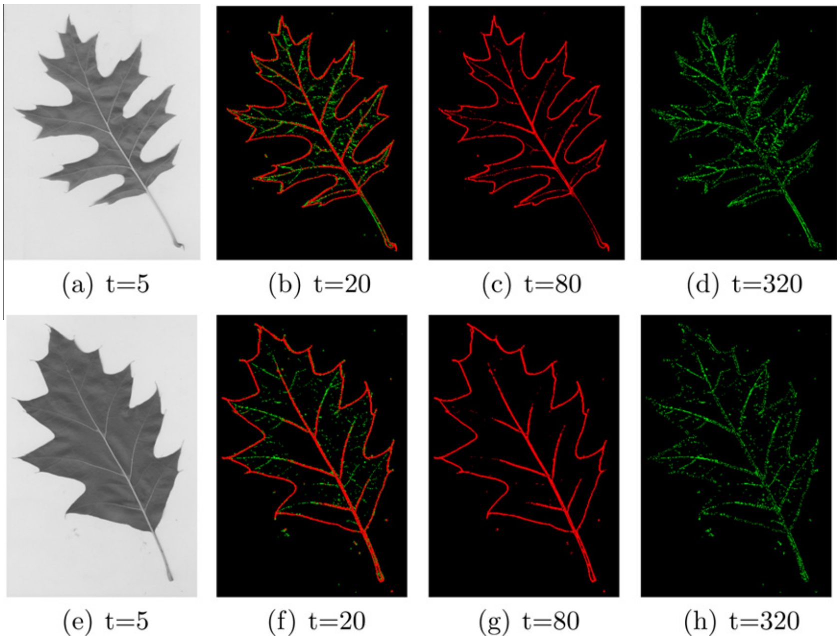
[2] It should be noted that the specifics of leaf venation types and orders are beyond the scope of this paper. Primary and secondary venations are viewed in this paper from an image processing perspective, with only limited alignment to any specific botanical meaning. See for example (Runions et al., 2005) for a more in depth consideration of venation types.

[3] For interpretation of color in Fig. 9,15, the reader is referred to the web version of this article.


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

(e) (f) (g) (h)

Fig. 8. Example results comparing methods: original images, (a,e), ant algorithm output, (b,f), Sobel output, (c,g), Canny output (d,h). Please refer to a digital copy for clear images.


(a) t=5 (b) t=20 (c) t=80 (d) t=320

(e) t=5 (f) t=20 (g) t=80 (h) t=320

Fig. 9. Examples of using dual swarms with alternative thresholds to simultaneously evolve pheromone maps of different features. (a,e) Original images; (b,f) combined pheromone maps; (c,g) pheromone map from swarm with $T = 20$; (d,h) pheromone map from swarm with dual threshold $T_{upper} = 25, T_{lower} = 15$. Please refer to a digital copy for clear images.

Fig. 10 shows an example of using the above red weighted RGB heuristic information to allow the swarm to self-organise in response to red features in the image.

Fig. 11 shows the results of two swarms searching the same image, with one swarm using RGB red weighted heuristic information, as defined above, and the other using RGB blue weighted heuristic information, i.e.

$$\eta_{ij} = (B_{ij} - R_{ij}) + (B_{ij} - G_{ij}) \tag{14}$$

In Fig. 11(b) the red and blue pixels represent the positions of the swarm programmed with RGB red and blue weighted heuristic information, respectively. Likewise, in Fig. 11(c) the red and blue show the evolved pheromone maps from the two respective swarms. We can see how each swarm colony has converged onto the red and blue features within the image (in this case two parked cars), thus effectively resulting in a dual segmentation image represented by the evolved pheromone maps. In such a case, the two swarms are not interacting with one another, and are free to occupy the same pixels and move across one anothers paths.

5.2. Swarm COG daemon action

Previously the daemon action was defined such that any ant agents with $\eta_{ij} < T$ for more than Z consecutive time steps would

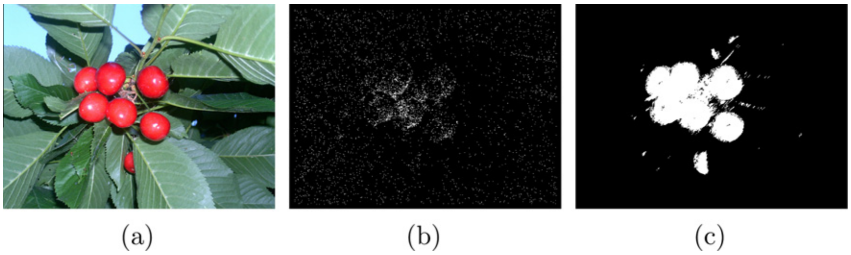


Fig. 10. An example of using RGB heuristic information for colour feature extraction, setting the visibility as a weighted average of the RGB red channel. (a) Original image; (b) agent positions; (c) pheromone map. Please refer to a digital copy for clear images. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

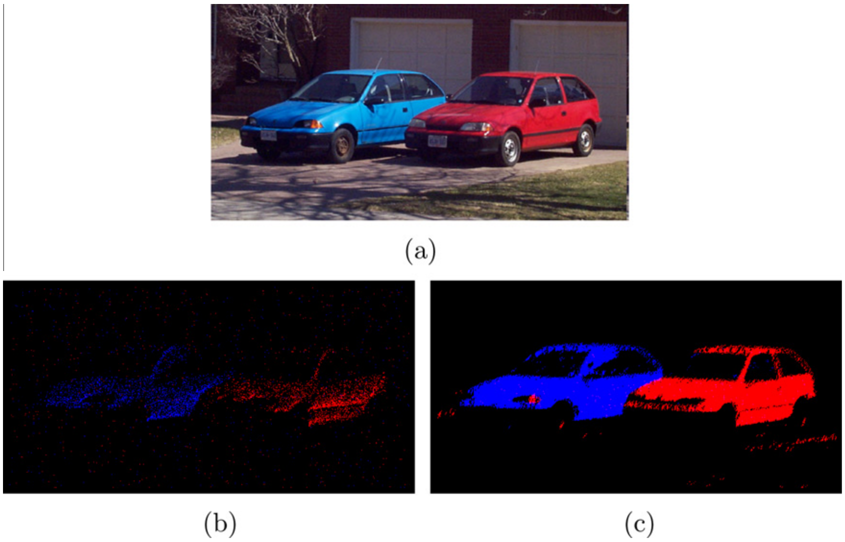


Fig. 11. An example of using two swarms simultaneously to extract different features. One swarm has visibility as a weighted average of the RGB red channel, and the other with the blue channel. (a) Original image; (b) agent positions; (c) pheromone map. Please refer to a digital copy for clear images. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

be terminated and replaced at a new 'random' location. For cases where the desired features are expected to be present throughout the whole image, as is often the case with image edge features (as used for the leaf pattern extraction), this set-up works well to expedite the process of self-organisation of the swarm towards the locations of the desired features.

For the application of feature extraction there are instances where we wish the ant colony to converge onto a single specific area of the image, i.e. a feature in the image that appears only once, in a single area of the image. In such a case, a change to the daemon action can act as a better catalyst to the self-organisation process.

After each agent has moved, the Centre Of Gravity (COG) of the swarm within the image is calculated as,

$$x_{COG} = \sum_n x_n/N, \quad y_{COG} = \sum_n y_n/N \tag{15}$$

where x_n and y_n are the x and y coordinates of the nth ant.

A condition is then set such that as long as $\eta_{x_{COG},y_{COG}} \geqslant T$ then the terminated agent is immediately replaced at the new location of (x_{COG}, y_{COG}), else, the terminated agent is immediately replaced at a new 'random' location, as before.

Fig. 12 shows results for the same set-up as for Fig. 11, except this time using the COG daemon action. We observe better convergence of the two swarms onto the red and blue cars, respectively, with less 'outlier' agents searching undesired areas of the image. This behaviour is as expected, and highlights how such modifications to the algorithm can alter the global behaviour of the swarms. It should be noted however that there is little difference between the two output pheromone maps, as even with the many outlier agents in Fig. 11(b), the swarm is large enough that there are enough agents over the desired regions to produce the desired pheromone map.

In a similar way, Fig. 13 shows an example of two swarms self-organising to different feature types in an artificial image comprising of a grid pattern and a solid red square. One swarm is using RGB red weighted heuristic information, and the other using grey-scale gradient information (as was used for the leaf venation extraction), also using the COG daemon action. Again we see how the red swarm has converged onto the red area of the image, with the green swarm mapping the prominent edge features.

6. Distributed adaptive threshold

The threshold value T serves two purposes. (i) To only allow ants to deposit pheromone if they are following an edge of gradient greater than T. This affects the self-organisation process by eliminating pheromone information for particularly weak edges (below T), and moreover, affects the algorithm output (i.e. the pheromone map) directly. (ii) To define the daemon action condition, replacing ants 'lost' searching areas of the image with visibility below T. This acts as a catalyst to the self-organisation process.

It follows that varying T will have a significant affect on the performance of the algorithm, both in terms of the output specific to

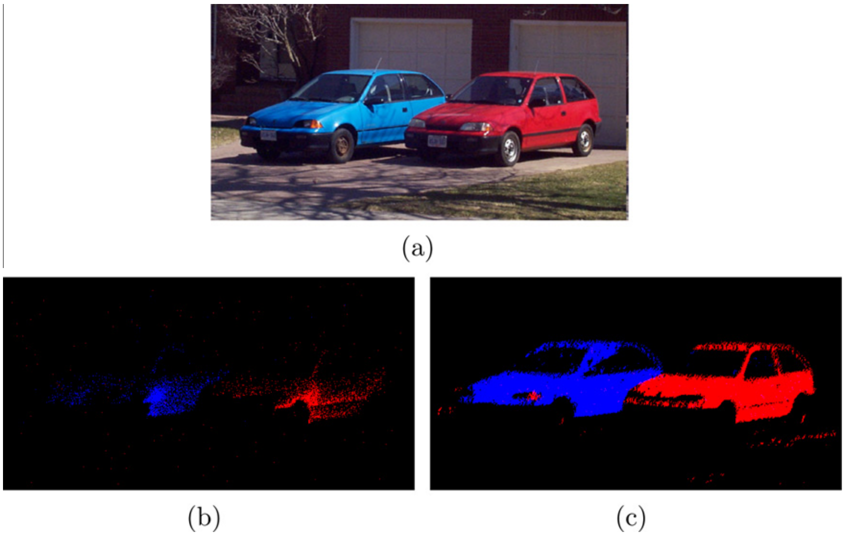


Fig. 12. An example of using two swarms simultaneously to extract different features. One swarm has visibility as a weighted average of the RGB red channel, and the other with the blue channel. This time using the COG daemon action. (a) Original image; (b) agent positions; (c) pheromone map. Please refer to a digital copy for clear images. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

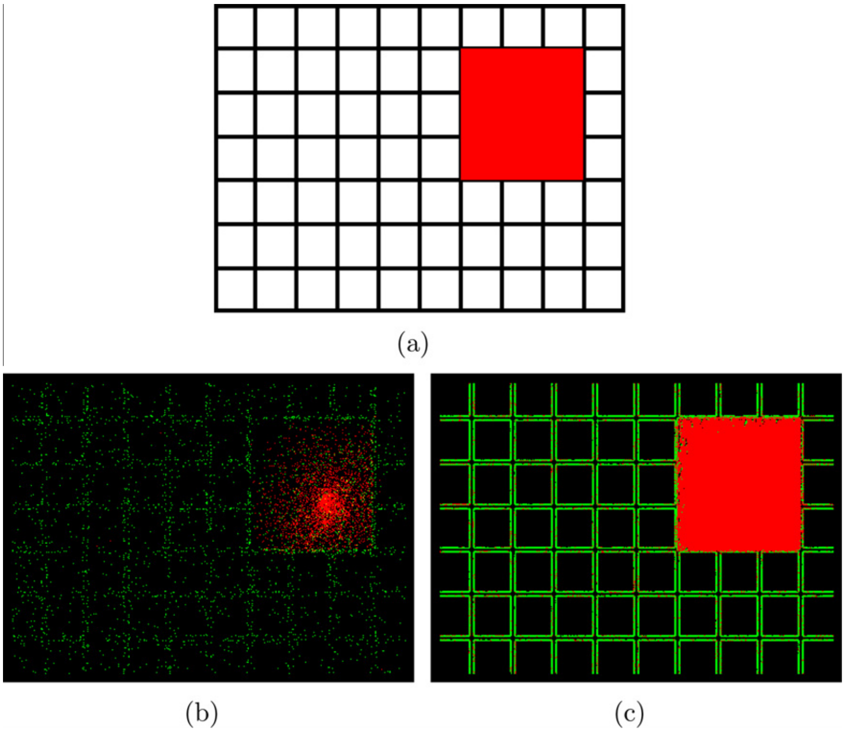


Fig. 13. An example of using two swarms simultaneously to extract different features from an artificial image, showing multiple feature self-organisation. One swarm has visibility as a weighted average of the RGB red channel, and the other with the image gradient heuristic information. (a) Original image; (b) agent positions; (c) pheromone map. Please refer to a digital copy for clear images. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the problem of feature extraction, and the general self-organisation process.

As expected, in Fig. 14(a) we observe a significant variation in Accuracy with varying *T*. Initially, as *T* increases, so does the resultant accuracy. This is then followed by a decrease in accuracy with further increase in *T* from *T* = 30 to *T* = 50. Again this is not unexpected given the mechanics of the algorithm, and the specific problem of edge feature extraction, giving rise to an optimum value of *T*, which will indeed be different depending on the characteristics of the specific image. It is interesting to note that while there is a significant variation in the resultant accuracy, the variation in *T* does not significantly affect the convergence rate.

In Fig. 14(b) we see an inverse relationship between *T* and average visibility, with increasing *T* resulting in lower average visibility for large *T*. This can be accounted for by considering the fact that the visibility is not directly affected by the threshold *T*. The self-organisation process is however affected by the daemon action, as well as the pheromone map, both of which are directly related to *T*. Increasing *T* to a large value that leaves only a small proportion of the image space above the threshold will have a detrimental

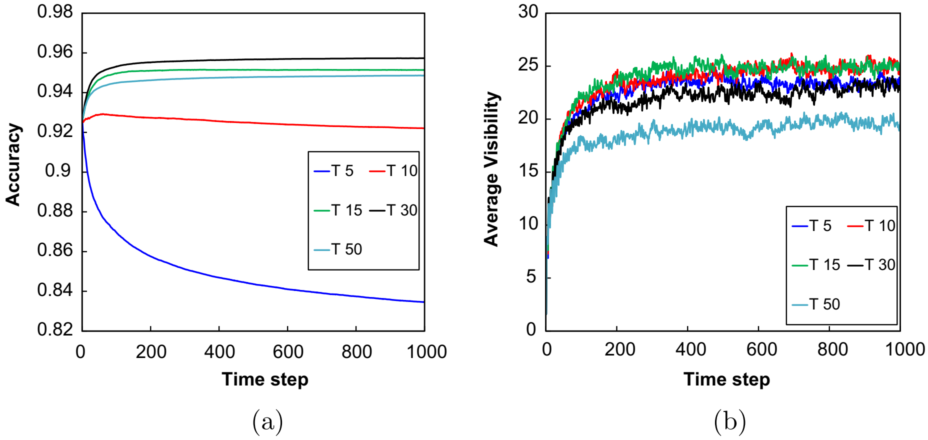


Fig. 14. Results for varying the threshold value T, applied to a typical real leaf image. Plots show the accuracy, (a), and average visibility, (b), versus time-steps.

effect on the self-organisation process, as the daemon action will be continuously replacing the majority of ant agents to new 'random' locations. With the majority of ant agents not converging on areas of high visibility, the average visibility will be lower.

Due to the threshold imposed on pheromone deposition in Eq. (10), and the conversion to a binary feature map in Eq. (11), a low threshold produces a low accuracy due to the small amounts of pheromone being deposited in the large regions of the image where there are only very weak edges present. These very weak pheromone trails will still be treated as image features, thus producing a binary feature map such as the example given in Fig. 15(a). A low or non-existent threshold will also mean that the daemon action will not function, thus leaving more ant agents searching the featureless regions, which increases the accuracy degradation further still. If we study τ, the pheromone map prior to conversion to binary (Fig. 15(b)), we can see that the pheromone trails along the prominent edge features are more pronounced. However in order to create a useful feature map, we would have to impose a specific threshold when converting to binary. This illustrates the requirement for the threshold T, in either way of implementation. Fig. 15(d) shows the final pheromone map, τ, again with $T = 1$ but with a higher evaporation rate, with $\rho = 0.02$. From visual inspection we see a much cleaner extraction of the edge features, with little noise present. If we inspect the corresponding τ_{BW} (Fig. 15(c)) we see there is still in fact a large amount of pheromone present in the background areas of the image, resulting again in a highly over segmented binary image.

The choice of threshold parameters is often critical in algorithm performance, and this case is no exception. Although it has been shown that the swarm can effectively evolve a pheromone map representing the pattern we wish to extract, a threshold is still required in order to remove the unwanted background information. In the next section, attention will be given to the concept of adaptive thresholding, where in the case of the developed ant algorithm, each individual agent will be able to adjust their own local thresholds as they move around the image.

6.1. Adaptive threshold

It was shown that using different values of T can significantly affect the performance of the algorithm, with a particular value of T being the optimum. Since finding the optimum, or even a satisfactory value of T can be very time consuming, attention is focused here on the use of an automated adaptive threshold. The idea of the adaptive threshold is to remove the requirement of a user-set threshold, which would otherwise need to be manually selected for different datasets and/or specific images.

The approach taken here focuses on exploiting the distributed nature of the ant-algorithm. Instead of employing a global threshold (where all ant agents use the same value of T), each of the $n \in N$ individual agents are allowed to maintain their own threshold value T_n. This approach is useful for dealing with image environments where different areas of the image have features of interest requiring different threshold values (for example a scanned image with varying light intensity levels, or the case of previous sections where primary and secondary leaf venations in scanned leaf images required different threshold values). Adapting individual agent's parameters that affects the overall global behaviour of the swarm is analogous to adaptation techniques used by swarms in nature, such as the previously mentioned honeybee's adapting their individual methods of communication to change group level foraging (Rozenberg et al., 1999).

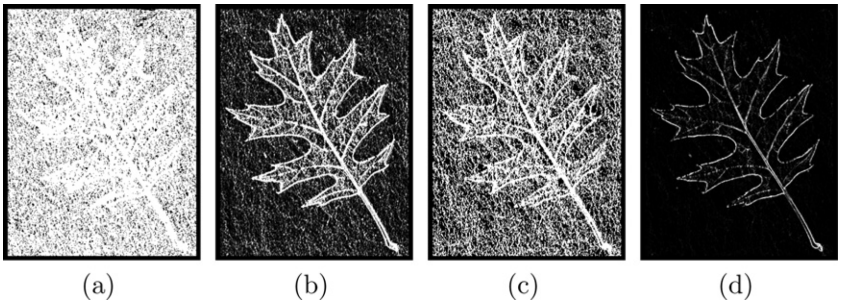
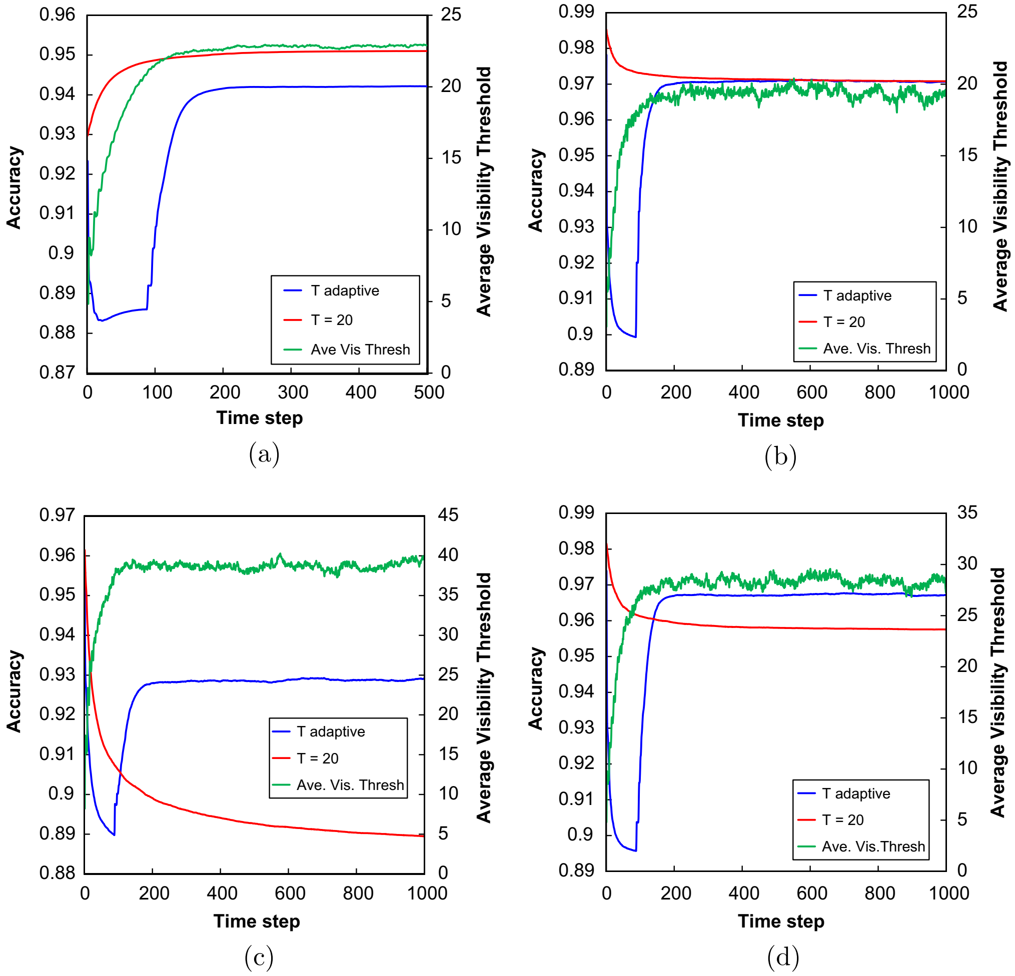


Fig. 15. A comparison showing the evolved pheromone map with $T = 1$, (a) the standard binary feature map τ_{BW}, (b) the final pheromone map τ, prior to binary conversion, (c) τ_{BW} with a greater evaporation rate of $\rho = 0.02$, and (d) τ with $\rho = 0.02$. Please refer to a digital copy for clear images.



(a)

(b)

(c)

(d)

Fig. 16. A comparison of the accuracy versus time-steps, using the distributed adaptive threshold, and global static $T = 20$ separately. (a) Averaged over the 20 images from the RBG KEW leaf data-set used previously; (b) for the image in Fig. 17(a); (c) for the image in Fig. 17(d); (d) for the image in Fig. 17(h). The plots also include the average visibility threshold of all the agents (when using the distributed adaptive threshold), showing how the threshold adapted over time.

Since the ants are (at least initially) distributed throughout the image environment, each individual agent sets its initial T_n value according to its local neighbourhood. The adaptive threshold is initialised such that, for agent n, at timestep $t = 1$,

$$T(1)_n = \sum_{S=1}^{8} \eta_S/8 \tag{16}$$

where η_S here is the visibility of the 8 surrounding pixel locations.

This allows an automated initial ‘first guess’ at the threshold value. For subsequent time-steps the adaptive threshold is defined by the following update rule:

$$T(t)_n = \begin{cases} T(t-1)_n + 1 & \text{if } \eta_n(x,y) > T(t-1)_n \\ T(t-1)_n - 1 & \text{if } \eta_n(x,y) < T(t-1)_n \\ \overline{T} & \text{if } T(t-1)_n < \overline{T} \end{cases} \tag{17}$$

where $\eta_n(x,y)$ is the current visibility of agent n at pixel location (x,y), and $\overline{T}$ is the average threshold over all agents.

This novel update rule allows individual ant agents to increase or decrease their individual threshold values depending on their individual visibility perception where they are in the environment, while maintaining a threshold that is above the level of background noise, as calculated from the global perception of the entire swarm.

Fig. 16(a) compares results averaged over the 20 leaf image test dataset used in the case-study above. The plot shows the accuracy versus timesteps for the previously used global $T = 20$ (determined by trial and error to give good results across the leaf dataset), and for the adaptive threshold method, $T_{adaptive}$. The plot also shows how the average threshold value $(\sum T_n/N)$ varies over time. The accuracy here is the statistical accuracy calculated by comparison to ground truth images (as defined in the previously). We see that the $T_{adaptive}$ method has resulted in less accuracy than for the static $T = 20$ threshold, which was chosen by systematic trial and error. This suggests that the adaptive method is not finding the optimum value of T. However, from qualitative visual inspection, the results obtained are still of a good quality (with a statistical accuracy of approximately 0.009 less than with $T = 20$). We see that on average, the average threshold converges to a value of approximately $\bar{T} = 23$, which is close to the value obtained via systematic trial and error.

Fig. 16(b) shows a similar plot for the image shown in Fig. 17(a)[4]. We notice how the accuracy for the adaptive threshold and the static threshold of $T = 20$ have converged to approximately the same value in this particular case. It is interesting to note that the average visibility threshold has converged close to $T = 20$, which would account for the similar performance in this case. Fig. 16(c) shows results for the image in Fig. 17(d), this time showing the adaptive threshold method to yield higher accuracy. If we draw our atten-

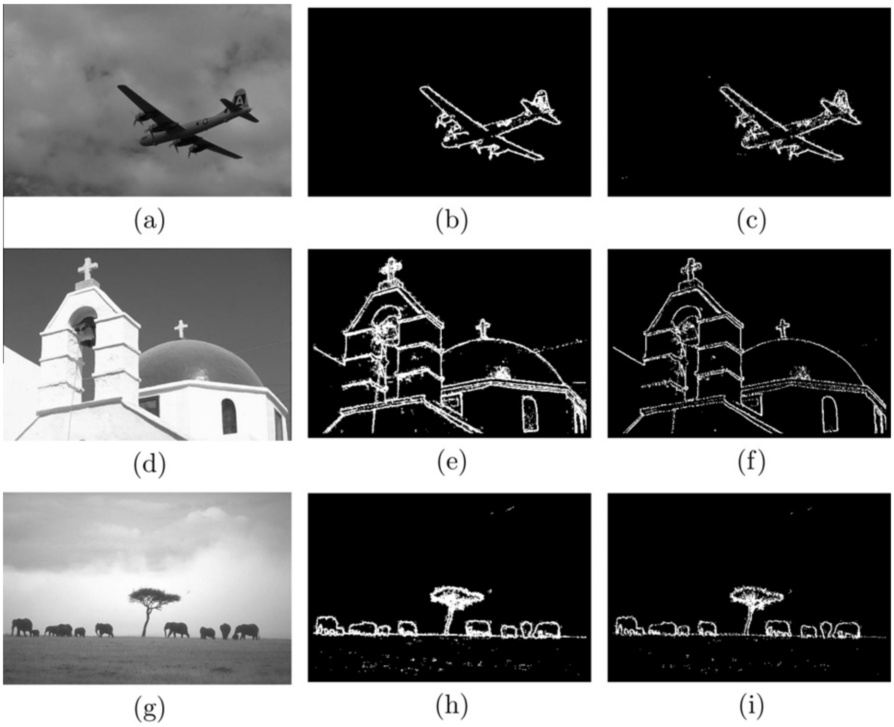


Fig. 17. Example images (a, d, g) showing the evolved pheromone map using the distributed adaptive threshold (c, f, i), comparing the evolved pheromone map with global static $T = 20$ (as used previously for the leaf image data-set) (b, e, h).

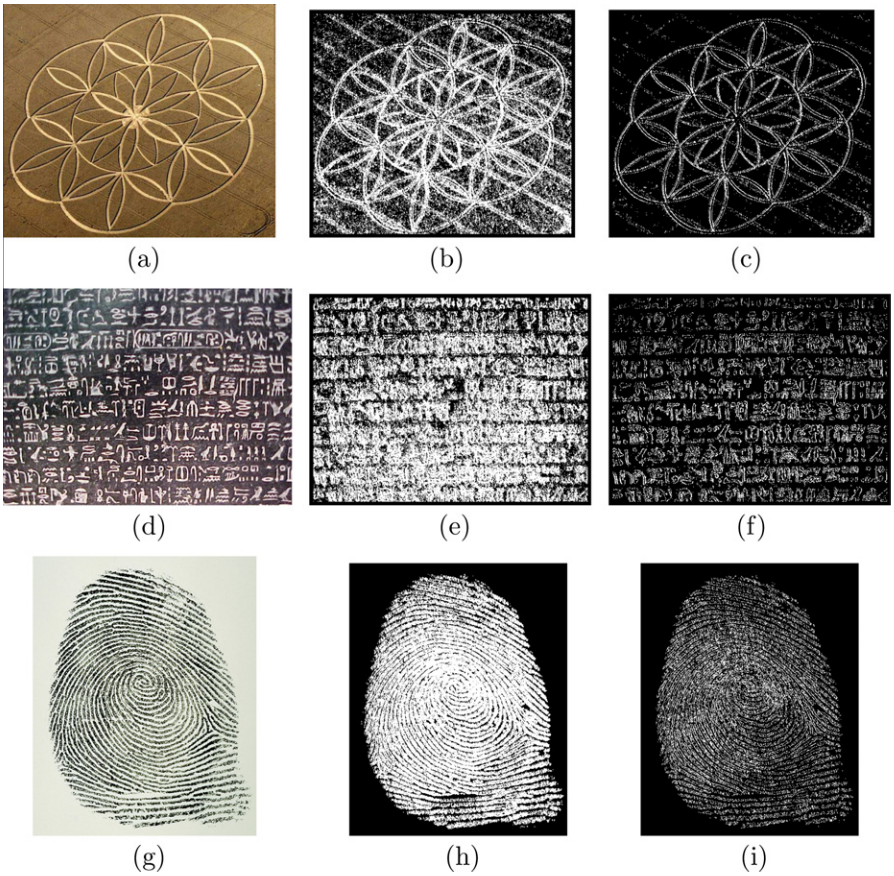


Fig. 18. Example images (a, d, g) showing the evolved pheromone map using the distributed adaptive threshold (c, f, i), comparing the evolved pheromone map with global static $T = 20$ (as used previously for the leaf image data-set) (b, e, h).

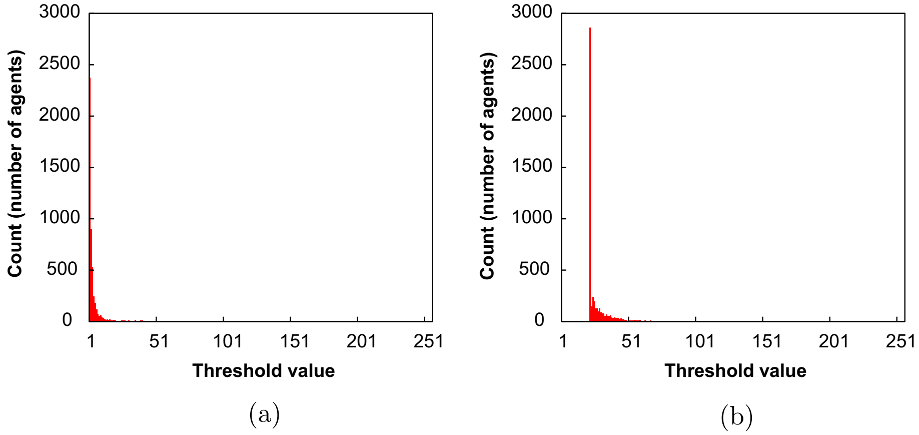


Fig. 19. Histogram plots showing the distribution of individual threshold values in the swarm population, (a) at the beginning of the algorithm run ($t = 1$), and (b) at the end of the algorithm run ($t = 1000$), for a typical leaf image.

tion to the corresponding evolved pheromone maps (Fig. 17(f) and (g), for $T = 20$ and $T_{adaptive}$, respectively), we observe over segmentation for the case of $T = 20$, with $T_{adaptive}$ suffering less from this problem. Fig. 16(c) shows the average visibility threshold to converge close to $T = 40$ in this case, which accounts for the difference in accuracy from the different instances, and the fact that the $T = 20$ instance resulted in over segmentation. We see similar, less pronounced results in Fig. 16(d) and Fig. 17(h–j), again with the $T = 20$ instance showing greater over segmentation.

It should be noted that in Fig. 16(b–d), the accuracy for the $T = 20$ instances decrease with time (in contrast to Fig. 16(a)). This is because in the ground-truth data for these images the edge features are defined as 1 pixel width boundaries (as opposed to the 3 pixel width used in the leaf image ground truth data). Along with the already high true negative count, this results in a particularly high accuracy to start with. For this comparison, of most interest however is the final accuracy, once the swarm has converged, as a direct comparison of performance. The nature of the curve is of

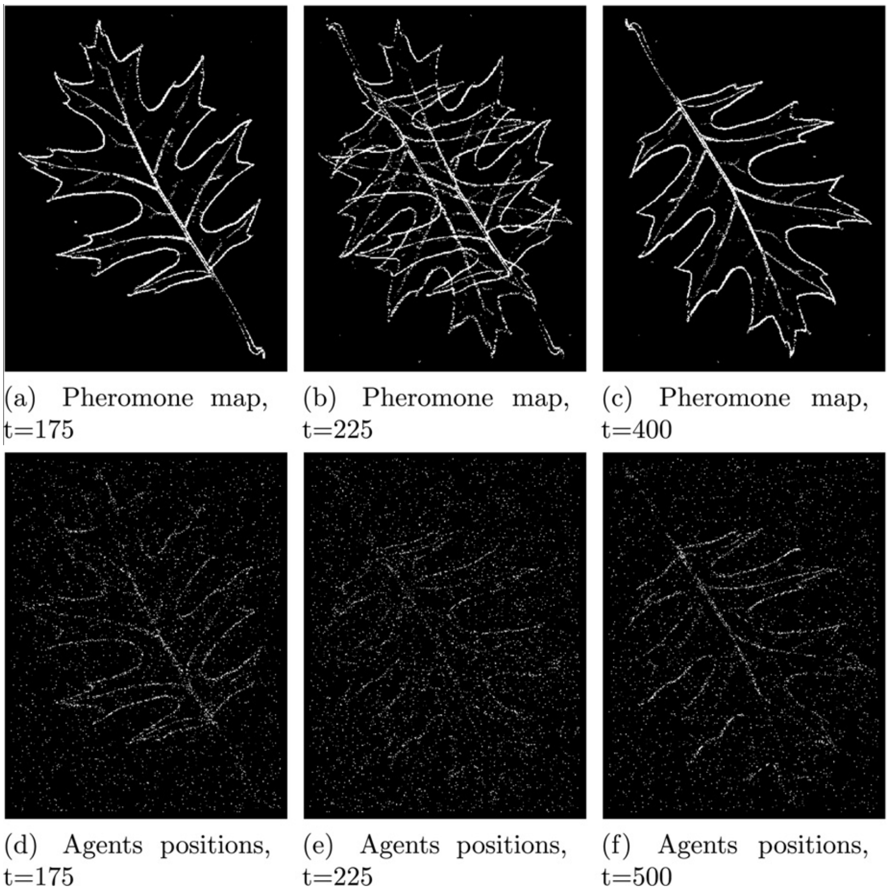

(a) Pheromone map, t=175
(b) Pheromone map, t=225
(c) Pheromone map, t=400
(d) Agents positions, t=175
(e) Agents positions, t=225
(f) Agents positions, t=500

Fig. 20. An example of the swarm adapting to a change in environment structure, showing the evolving pheromone map (a–c) and agents positions (d–f), at different time-steps. At $t = 200$ the image environment is rotated 180 degrees clockwise. Please refer to a digital copy for clear images.

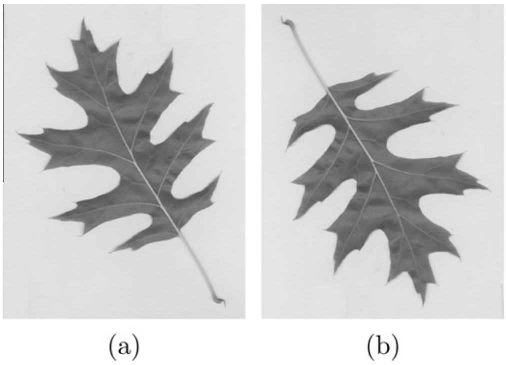


Fig. 21. (a) Initial and (b) rotated image environment.

interest with respect to analysing the self-organising behaviour of the swarm (i.e. the convergence rate).

Analysing the curves for the adaptive threshold accuracy, we can assess the different self-organisation process that is occurring with this method. The initial drop in accuracy seen with this method results from the fact that in the early time-steps the majority of the agents will have a low threshold (as can be seen with the corresponding average visibility threshold curves), due to many of the agents starting in 'background' regions of the image. This results in a rapid build-up of pheromone in the background regions of the image, thus reducing the statistical accuracy. As the swarm begins to converge onto the edge regions, simultaneously the pheromone in the background regions will begin to evaporate, and this is where we observe a rapid increase in the accuracy until the swarm converges and the accuracy curve levels out.

Fig. 18 shows additional example comparison images of the evolved pher- omone map obtained using $T = 20$ and $T_{adaptive}$, for a range of example images. Qualitatively, from visual inspection we see that in these cases, the threshold of $T = 20$ produces an over evolved pheromone map (or over segmented image), with the adaptive threshold producing more visually concise results, qualitatively better representing the pattern of edge features in the images. These are examples of where, in the manual case, one would have to go through the often lengthy process of trial and error again, in order to find the optimum value of T for these different images. Using $T_{adaptive}$ however does not require this process, and although it may not find the optimum threshold, it does produce near optimum results.

Fig. 19 shows the distribution of the individual agents threshold values at the beginning and end of an algorithm run for a typical leaf image, using $T_{adaptive}$. We can see from the two histograms that the distribution has effectively shifted from a majority population with a threshold value of $T = 1$ at the beginning, to a majority population with a threshold value of $T = 22$ at the end. This is consistent with the majority of the swarm starting on background areas of the image, thus from Eq. (15) the threshold will be relatively small owing to only minor changes in gradient on the background areas of the image. As the swarm self-organises onto the

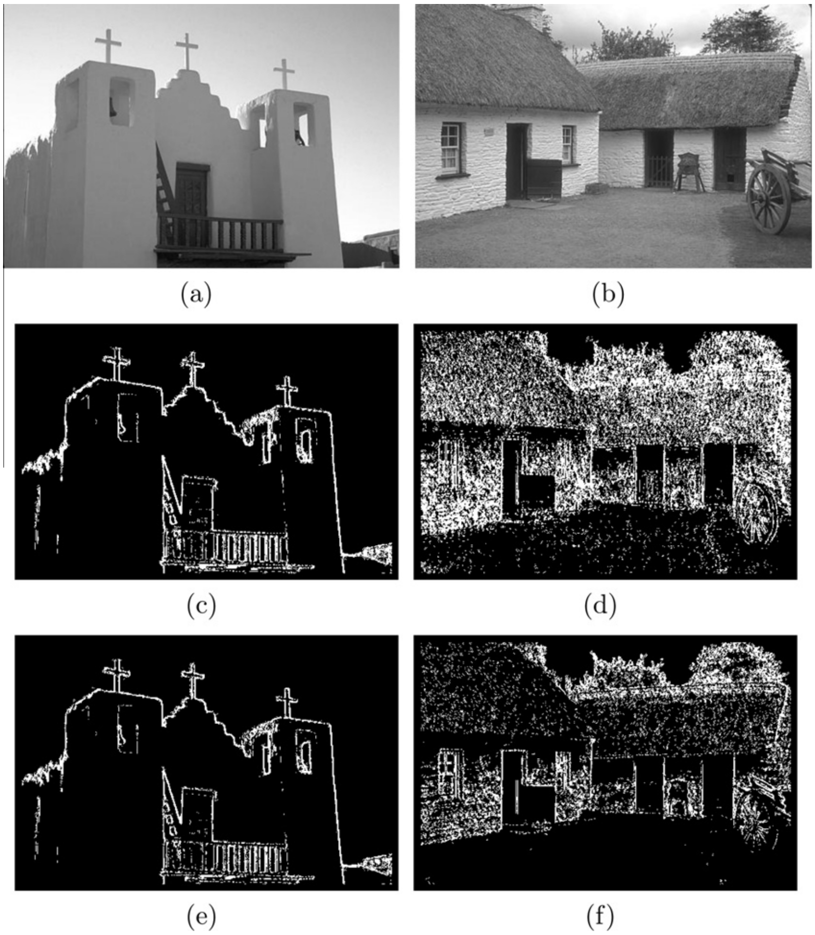


Fig. 22. An example of the swarm adapting to a change in environment structure, comparing the algorithm with global static $T = 20$ ((c) at $t = 495$, (d) at $t = 1000$) and $T_{adaptive}$ ((e) at $t = 495$, (f) at $t = 1000$). The image is swapped from (a) to (b) at $t = 500$.

edge regions of the image, the agents adjust their threshold values accordingly, giving rise to the shift in threshold population we see in Fig. 19(b).

It is clear that different images will inevitably require different thresholds in order to produce the desirable segmentation/edge map. Inspired by swarm adaptation in the wild, the adaptive method presented here, although it does not guarantee the optimum results, it does provide acceptable results, without the requirement for any threshold. This is achieved through exploiting the distributed nature of the ant-algorithm approach, and the self-organising nature of the embedded swarm. The parameter adaptation here is effectively allowing the swarm to learn appropriate threshold values for a given image environment. The adaptation process occurs simultaneously with the pattern self-organisation of the swarm, complimenting the temporal nature of the ant-algorithm, and this approach to the specific image processing problem setting.

7. Pattern adaptation

The self-organisation methods of the ant-algorithm approach, and indeed of the swarm intelligence paradigm in general, are inherently adaptive. This section focuses on further exploiting such adaptive nature for the purpose of pattern adaptation, again employing the ant-algorithm approach in digital image environments.

To study the swarms ability to adapt to changing environment structures we can consider the problem of allowing the swarm to self-organise to a particular static image, and then change the environment to another image, and observe the adaptation process. Fig. 20 shows an example of the swarm adapting to a changing image environment. From $t = 1$ to $t = 199$ the image environment is that of Fig. 21(a), then at $t = 200$ the image environment is switched to that of Fig. 21(b).

In Fig. 20 the adaptation process is visualised by showing the changing pheromone map and agents positions. Fig. 20(a) and (d) show the evolved pheromone map and agents positions respectively, at $t = 175$, as the swarm has adapted to the original image (Fig. 21(a)). Fig. 20(b) and (e) show the same at $t = 225$, shortly after the image environment has been rotated 180 degrees clockwise (Fig. 21(b)), and Fig. 20(c) and (f) show the same at $t = 400$, after the swarm has re-adapted to the new image environment. From visual inspection we can see how the swarm has adapted to the change in environment, re-adapting to the new image pattern. Fig. 20(b) shows the new pattern emerging in the pheromone map, while the old pattern is evaporating away.

Figs. 22 and 23 give further examples of pattern adaptation. From $t = 1$ to $t = 499$ the algorithm runs with images (a), then swaps to images (b) for $t = 500$ to $t = 1000$. Images (c) and (e) give the evolved pheromone maps at $t = 495$ for the algorithm run with a static $T = 20$ and $T_{adaptive}$, respectively. Images (d) and (f) show the same at time $t = 1000$. Corresponding plots of accuracy and average visibility threshold versus time-steps are given in Fig. 24.

Again in these two examples we observe that the pheromone maps have evolved to the edge feature patterns of the initial image, and then adapted to the second images, with the previous pattern being fully evaporated over time. From the accuracy plots in Fig. 24 we can see that for both examples, and both methods, the adaptation process takes approximately 200 time-steps. For the adaptive

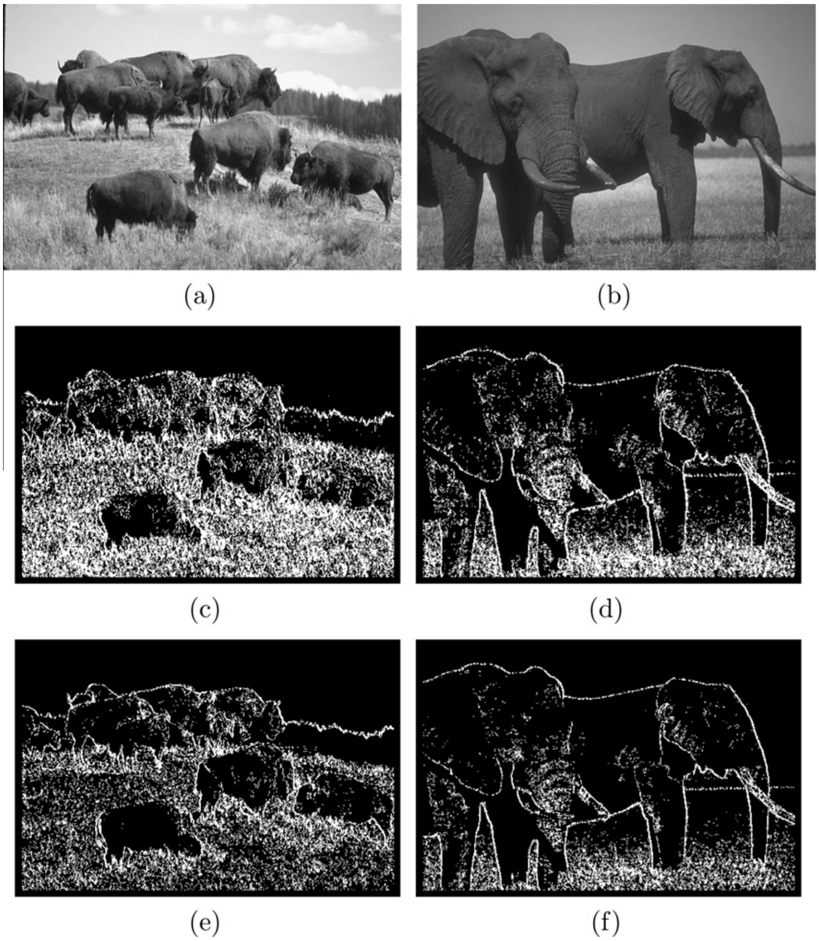


Fig. 23. An example of the swarm adapting to a change in environment structure, comparing the algorithm with static $T = 20$ ((c) at $t = 495$, (d) at $t = 1000$) and $T_{adaptive}$ ((e) at $t = 495$, (f) at $t = 1000$). The image is swapped from (a) to (b) at $t = 500$.

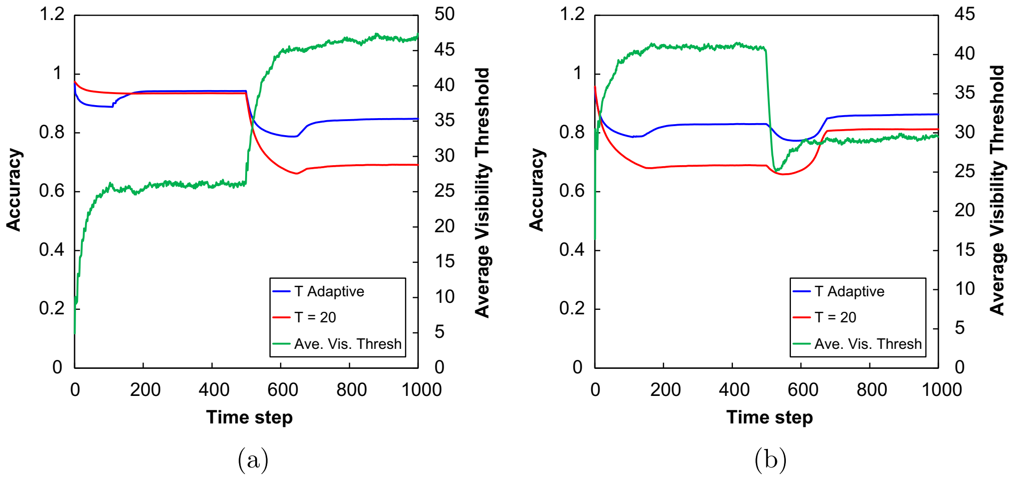


Fig. 24. A comparison of the accuracy versus time-steps, using the distributed adaptive threshold, and global static $T = 20$ separately. (a) Plot corresponding to Fig. 22, and (b) corresponding to Fig. 23. The plots also include the average visibility threshold of all the agents (when using the distributed adaptive threshold), showing how the threshold adapted over time.

threshold method, it is interesting to analyse the change in average visibility threshold of the swarm, during the adaptation process. The change in image for the example in Fig. 22 results in an increase in average visibility threshold, while for the example in Fig. 23 there is a decrease. In both cases the resultant accuracy for the adaptive threshold method is greater. The difference in converged accuracy between $T_{adaptive}$ and $T = 20$ in Fig. 24(a) for the initial image is small, as the average visibility threshold is close to 20, then as the image changes, and the average visibility threshold increases to approximately 47, the difference in accuracy also increases. For both methods there is a decrease in accuracy in the first example and an increase in the second, when the image is swapped. This is reflected in the visually over-segmented images in the emerged pheromone maps in Fig. 22(d) and (f) and Fig. 23(c) and (e).

The adaptive threshold is shown here to increase the adaptable capabilities of the swarm, by allowing the swarm to autonomously change a key parameter in response to a change in the perceived environment and adapt accordingly. This results in adaptation on two levels: (i) the swarm physically adapts its structure and movements by self-organising to the environment features, exhibiting *group level* adaptive pattern formation; (ii) individual agents adapt their threshold values in response to the change in image environment, thus adapting their response to the environment in terms of *individual* movement and pheromone deposition.

The physical adaptation is fundamental to the ant-algorithm design. The individual parameter adaptation augments the adaptive capabilities of the swarm allowing for autonomous, *in situ* distributed parameter self-optimisation.

8. Conclusions and discussion

This paper has investigated the self-organising properties of an ant-algorithm (based on the Ant System approach Dorigo & Gambardella, 1997) when placed in a digital, discretised environment, in the form of a digital image landscape. A new implementation of the Ant System algorithm has been developed, specifically tailored for image feature extraction.

As a novel application for ant-algorithms, a case-study in collaboration with RBG KEW for image feature extraction was carried out, where the algorithm was specifically set-up for the autonomous extraction of leaf outline and venation pattern, from digitally scanned images of live Quercus leaves. This case-study provided an application focus and example data-set in order to assess the performance of the algorithm for a specific feature extraction problem, and to study, in more general terms, the way in which a swarm of embedded software agents can form complex patterns in response to local environment stimuli.

The performance of the algorithm has been evaluated (in the context of the case-study feature extraction problem) by a qualitative analysis of the output images as well as a quantitative analysis by statistical measures against ground truth images. The analysis showed the algorithm to work well, producing acceptable statistical accuracy (measured against ground truth data), and visually accurate representations of the desired features. The quantitative analysis has improved on previous work in the area of ant-algorithms for image processing, by providing statistical results to assess the performance of this approach, as well as investigating the performance in relation to a novel real-world application.

A comparison to well known existing methods for edge feature extraction yielded comparative results in terms of average accuracy achieved over a sample dataset. The comparison to existing methods and quantitative exploration of the algorithm parameters has provided further contributions to knowledge in the research area of ant-algorithms for image processing.

This analysis has pointed out some limitations such as sensitivity to noise. This however is not specific to the proposed approach, and is in fact a typical problem amongst edge detection and feature extraction algorithms.

One of the main motivations behind the use of artificial swarms in the area of image processing is to improve on robustness and automation. Many traditional edge detection methods, such as the previously mentioned Canny edge detector (Canny, 1986), operate on the entire image in a linear fashion, and require a number of parameters to be tuned for a given image in order to produce satisfactory results. The ant algorithm approach works in a different way, relying on the phenomenon known as *stigmergy* (Bonabeau et al., 1999) to produce self emergent behaviour amongst the artificial swarm in response to the given environment, which in this case is a digital image. When programmed with a search preference towards high image gradient change, the swarm converges onto the stronger edges of the image and the resulting pheromone field produced by the swarm maps out these edges. This convergence is reflected in Fig. 4(a) with the rapid increase in accuracy over time during the early time-steps as the swarm is converging onto the image edges. After approximately 400 time-steps the gradient of the accuracy curve has almost dropped off to zero, as convergence is achieved.

The use of multiple swarms for multiple features is an interesting concept which deserves some discussion. Extracting multiple features in image processing is not a new concept, however using swarms in this context is. This method exploits the distributed temporal nature of the swarm approach to simultaneously evolve multiple feature maps and patterns from the image data by means of self-organisation of multiple swarms. This is similar to what has been observed in social insect colonies in nature, where instead of performing all the required tasks, a worker usually specialises in a set of tasks. This division of labour, whereby different tasks are performed simultaneously by groups of specialised individuals, is believed to be more efficient than if tasks were performed sequentially by unspecialised individuals (Bonabeau et al., 1999). By having multiple swarms using different heuristic information we are able to utilise this notion and observe simultaneous self-organisation in response to different features in a common digital image environment.

A potential advantage of the swarm intelligence approach is the potential to remove the requirement of a sensitivity threshold. In many traditional methods choosing an appropriate threshold value is of most importance in obtaining the best quality results, and indeed it has been shown that this is still the case with the ant-algorithm approach in its current state. This paper has extended the ant-algorithm approach by focusing on adaptation in the self-organisation process. The swarms abilities have been augmented by allowing individual ant agents to control their own threshold parameter settings. The way in which the agents adapt their parameters has been purposefully limited to simple rule-based logic, to maintain a level of simplicity akin to the local decision making processes observed in real-world swarms. Experimentation for parameter adaptation has focused on adapting the threshold value T, which was shown in previous sections to be of particular importance in determining the performance of the algorithm for the purpose of image feature extraction, and which is likened to the internal behavioural thresholds believed to be present in social insects in nature.

Despite these simple rules, it has been shown that while the swarm may not collectively converge on the optimum parameter setting for a given image, by allowing individuals to continually adjust their threshold T values, the average T value of the swarm does converge as the swarm converges, with the resultant evolved pheromone map showing the edge features as desired, over a range of different images with varying threshold values. From an image segmentation application point of view, eliminating the requirement for setting a threshold and still producing good quality results is a particular advantage. Setting an appropriate threshold requires prior knowledge of the image or images to be processed, and an often lengthly process is required to determine an appropriate threshold for each image set, or even each individual image, to be processed. The use of the ant-algorithm swarm approach with the proposed adaptive threshold provides a novel image and video processing solution which exploits the distributed nature of the swarm self-organising approach, to deliver distributed adaptive feature extraction and tracking in digital image media.

From a swarm intelligence theoretical standpoint, this reinforces the concept of nature inspired distributed simple decision making as a powerful optimisation and problem solving tool in the digital, computerised world. Giving individual swarm agents the facility to dynamically adjust their threshold parameter in response to the changing environment increases the self-organisation capability of the swarm by increasing adaptability. This allows the swarm to successfully self-organise into patterns representing the image feature structure over a wider range of image characteristics (for example ambient lighting, entropy, noise).

The self-organising pattern formation behaviour of the swarm has been further examined by considering dynamic imagery. In such a case, there is an additional challenge for the swarm; once a particular pattern formation has been reached, the swarm must be able to 'forget' this pattern formation and adapt, through continuous self-organisation, to a new pattern, or indeed a continuously changing pattern. The use of parameter adaptation with dynamic imagery has shown to yield promising results, with the swarm changing the threshold value in accordance with changing images, resulting in increased accuracy over an otherwise static threshold value.

This paper has attempted to examine a number of adaptive self-organisation methods, in terms of ant-algorithms for image processing applications. A level of success has been achieved in harnessing the power of distributed adaptation for the purpose of implementing an adaptive threshold technique for the presented ant-algorithm. Increasing the dynamic nature of the environment has introduced new challenges, and provided additional discussion of the inspiration and knowledge we can gain from swarms in nature, as well as the analogy between natural and computerised swarms.

References

Royal Botanic Gardens, KEW. (2011). <http://www.kew.org>.

Arcaute, E., Christensen, K., Sendova-Franks, A., Dahl, T., Espinosa, A., & Jensen, H. J. (2009). Division of labour in ant colonies in terms of attractive fields. *Ecological Complexity, 6*(4), 396–402.

Bonabeau, B., Dorigo, M., & Theraulaz, G. (1999). *Swarm intelligence: From natural to artificial systems*. New York, NY: Oxford University Press.

Calderone, N. W., & Page, R. E. (1996). Temporal polyethism and behavioural canalization in the honey bee, *Apis mellifera*. *Animal Behaviour, 51*, 631–643.

Canny, J. A. (1986). Computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 8*, 679–698.

Channa, A. H., Rajpoot, N. M., & Rajpoot, K. M. (2006). Texture segmentation using ant tree clustering. In *2006 IEEE international conference on engineering of intelligent systems* (pp. 1–6).

Chialvo, D., & Millonas, M. M. (1995). How swarms build cognitive maps. In L. Steels (Ed.), *The biology and technology of intelligent autonomous agents* (pp. 439–450). Springer-Verlag.

Chittka, L., & Muller, H. (2009). Learning, specialization, efficiency and task allocation in social insects. *Communicative and Integrative Biology, 2*(2), 151–154.

Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperating learning approach to the travelling salesman problem. *IEEE Transactions on Evolutionary Computation, 1*(1), 53–66.

Fernandes, C., Ramos, V., & Rosa, A. (2005a). Varying the population size of artificial foraging swarms on time varying landscapes. In W. Duch, J. Kacprzyk, E. Oja, & S. Zadrozny (Eds.), *Artificial neural networks: Biological inspirations – ICANN 2005. Lecture notes in computer science* (Vol. 3696, pp. 311–316). Berlin/Heidelberg: Springer.

Fernandes, C., Ramos, V., & Rosa, A. C. (2005b). Self-regulated artificial ant colonies on digital image habitats. *International Journal of Lateral Computing, 2*(1), 1–8.

Ma, L., Tian, J., & Yu, W. (2010). Visual saliency detection in image using ant colony optimisation and local phase coherence. *Electronic Letters, 46*, 1066–1068.

Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the 8th international conference on computer vision* (Vol. 2, pp. 416–423).

Mullen, R. J., Monekosso, D., Barman, S., Remagnino, P., & Wilkin, P. (2008). Artificial ants to extract leaf outlines and primary venation patterns. In *Proceedings of ANTS'2008 – 6th international workshop on ant algorithms*. Brussels, Belgium.

Nezamabadi-pour, H., Saryazdi, S., & Rashedi, E. (2006). Edge detection using ant algorithms. *Soft Computing, 10*, 623–628.

Ouadfel, S., & Batouche, M. (2002). Unsupervised image segmentation using a colony of cooperating ants. In *BMCV '02: Proceedings of the second international workshop on biologically motivated computer vision* (pp. 109–116). Springer-Verlag.

Ramos, V., & Almeida, F. (2000). Artificial ant colonies in digital image habitats – a mass behaviour effect study on pattern recognition. In *Proceedings of ANTS'2000 – 2nd international workshop on ant algorithms (From ant colonies to artificial ants)* (pp. 113–116), Brussels, Belgium.

Rauch, E. M., Millonas, M. M., & Chialvo, R. (1995). Pattern formation and functionality in swarm models. *Physics Letters A, 207*, 185–193.

Reznikova, Z. I., & Novgorodova, T. A. (1998). The importance of individual and social experience for interaction between ants and symbiotic aphids. *Doklady Biological Sciences, 359*, 173–175.

Rozenberg, G., Back, T., Blum, C., Eiben, A. E., Kok, J. N., Merkle, D., et al. (1999). *Swarm intelligence: Introduction and applications*. Berlin: Springer-Verlag.

Runions, A., Fuhrer, M., Lane, B., Federl, P., Rolland-Lagan, A., & Prusinkiewicz, P. (2005). Modeling and visualization of leaf venation patterns. *ACM Transactions on Graphics, 24*, 702–711.

Wilkin, P. (2008). *Personal communication*. London, England: Royal Botanic Gardens, KEW.