



WCIT-2010

## Extraction of flower regions in color images using ant colony optimization

Doğan Aydın<sup>a</sup>\*, Aybars Uğur<sup>a</sup>Ege University Computer Engineering Dept., Bonova 35100 İzmir, Turkey

### Abstract

Extraction of flower regions from complex background is a difficult task and it is an important part of a flower image retrieval and recognition. In this article, we propose an Ant Colony Optimization (ACO) algorithm as a general color clustering method, and test it on flower images as a case study of object boundary extraction. The segmentation methodology on flower images consists of six steps: color space conversion, generation of candidate color cluster centers, ant colony optimization method to select optimum color cluster centers, merging of cluster centers which are close to each other, image segmentation by clustering, and extraction of flower region from the image. To evince that ACO algorithm can be a general segmentation method, some results of natural images in Berkeley segmentation benchmark have been presented. The method as a case study on flower region extraction has also been tested on the images of Oxford-17 Flowers dataset, and the results have confronted with other well established flower region extraction approaches.

© 2010 Published by Elsevier Ltd. Open access under [CC BY-NC-ND license](#).

Selection and/or peer-review under responsibility of the Guest Editor.

**Keywords:** Ant colony optimization, color image segmentation, flower region extraction

### 1. Introduction

Segmenting image in a separate regions and finding relationships among them are needed to understand an image. The task of extracting image into the meaningful regions and objects is referred as image segmentation. Image segmentation is a complex and hard task in color images, but also it is the one of the important and curial problems in color image analysis. Because high performing segmentation algorithms lead effective image recognition and retrieval systems.

Color image segmentation algorithms in literature depend on whether the distribution of pixel color is analyzed in the image plane or in a color space [1]. Techniques that use image plane are usually based on region growing [2] or split and merge strategy [3]. Other techniques which are based on color space analysis first try to determine best color space, and then divides image into color classes which are constructed by an analysis of color histogram [4-6] or by a cluster analysis procedure [7-9].

In this article, we propose an Ant Colony Optimization (ACO) approach to color image segmentation by using color clustering and we test in a specific domain of object boundary extraction: flower region extraction. The method tries adaptively to determine number of color clusters and cluster center points optimally, and assigns pixels to their nearest clusters. Then merging mechanism steps in, and it merges closest clusters according to the color

\* Doğan Aydın. Tel.: +90-232-3434000-5337; fax: +90-232-339940

E-mail address: [dogan.aydin@ege.edu.tr](mailto:dogan.aydin@ege.edu.tr).



distribution in image. The main contributions of this paper are the following. First, purely ant colony optimization technique is applied on color image segmentation and flower region extraction at the first time. Ants in colony assist not only determining good cluster centers in image, but also selecting optimum cluster number in image adaptively. The second contribution is high performing method itself; in fact, experimental results show its efficiency in both general color image segmentation and object of interest segmentation in a specific domain.

The rest of this paper is organized as follows. In section 2, we describe Ant Colony Optimization (ACO). In section 3, we present our segmentation method step by step. Experimental results on natural and flower images are presented in section 4. Finally, in Section 5, we present some conclusions and directions for future work.

## 2. Ant Colony Optimization

Ant Colony Optimization (ACO) is a population-based meta-heuristic inspired from food foraging behavior of real ant colonies [10]. While foraging, ants deposit some amount of pheromone (a chemical substance) on their paths related to the food quality and amount, if they find some foods on their paths. Pheromone on environment informs other colony members, and reinforces ants to follow towards pheromone paths with a probability. The more pheromone on the path means higher probability to search around that path by the followers. Moreover, pheromone substance has evaporation behavior. Therefore, some paths may disappear because of pheromone evaporation if no ant follows on these paths. This evaporation behavior assists ants to forget long or bad paths, and finally only better and shorter paths become available to follow.

This trail-laying and trail-following behavior of ant colonies exhibits a probabilistic optimization technique to improve search direction in real world. In artificial and computer environment, ACO doesn't use directly real ant colony behavior but inspires. In ACO, artificial ants have local knowledge (heuristic) about the environment, and they act with a fix number of steps in a discrete time, contrary to real ants do.

Each artificial ant in ACO tries to find a good solution on search space (a discrete environment) by wondering step by step on search space states. At each step of solution construction of an ant, the ant selects a next suitable state and moves from one state to another according to a state transition probability function. This state transition probability between state  $i$  to  $j$  is determined as follows:

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

where  $\alpha$  and  $\beta$  are two adjustable constant parameters,  $\tau_{ij}$  is trail level of pheromone between state  $i$  and  $j$ ,  $\eta_{ij}$  is a heuristic function, and  $\Omega$  is set of the unvisited states.

Instead of pheromone depositing while solution construction, artificial ants deposit pheromone on their solution paths, once all tour constructions are completed by ants (at the end of iteration). Pheromone evaporation mechanism is also applied at the same time with pheromone depositing. This pheromone depositing/evaporating step is stated as *pheromone updating*, and generally it is applied on each step of the solution paths of ants according to the following formula:

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

where  $\rho$  is pheromone decay constant,  $m$  is the number of ant and

$$\Delta \tau_{ij}^k = \begin{cases} f_k / Q & \text{if } (i, j) \in T_k \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where  $f_k$  is the fitness function value,  $Q$  is a parameter which is usually set as 1 and  $T_k$  is the trip of the  $k^{\text{th}}$  ant.

Pheromone updating according to the solution quality leads ants to search new better solution around good solutions for the successive iterations, and by this way, solution quality increases during algorithm execution. When algorithm reaches the termination condition (which can be the maximum number of iteration or time), the best-so-far ant path is selected as a solution of the algorithm.

There are many successful works which are using ACO techniques to tackle many optimization problems (see the

lists of the problem to be tackled in [10]), but there are really few works in solution of segmentation and object detection in color images. Arney et al. [11] used ACO to tackle non-structured road boundaries detection problem on HSL (Hue-Saturation-Lightness) color space. Ant-based vessels detection in retinal image algorithm was applied by using only surface of green channel of RGB (Red-Green-Blue) images [12]. There is only one ACO-based work which is used for color clustering and color image segmentation [13], but this approach is not a purely ACO method but it is hybridized with Fuzzy-C Means algorithm.

### 3. The Proposed Methodology

To segment image into regions, our method operates some successive tasks step by step. First, it determines candidate cluster center points on which ant colony members wander step by step to construct solutions, and the heuristic values of cluster points are assigned. Then, ant colony optimization algorithm is run to select optimum cluster points over candidate cluster center points set. ACO may determine the solution in which some cluster center points may be closer to each others. To overcome this problem, closest cluster centers are merged. Finally, pixels are classified according to their closest cluster center point, and image is segmented into homogeneous regions. These steps are explained at the following subsections.

#### 3.1. Determining candidate cluster center points and their heuristic values

Before applying ACO method to image segmentation, at first, we have to define the environment and its states in which ant colony is living and tries to find a solution. To do this, color space is converted from RGB to CIE  $L^*a^*b$  color space where color distance between to pixels is easily determined by Euclidian distance measure. Then, histograms of each color space components are created and center points of bins are gathered (See an example in Figure 1). Let say that these bin centers sets are  $L = \{l_1, l_2, l_3, \dots, l_n\}$ ,  $a = \{a_1, a_2, a_3, \dots, a_n\}$  and  $b = \{b_1, b_2, b_3, \dots, b_n\}$  where  $l_i$ ,  $a_i$  and  $b_i$  are each bin centers and  $n$  is the number of bins in histograms. Then, candidate cluster center points set is determined from combination of these three bin centers sets. Hence, the candidates set (search space of ant colony) is defined as  $C = \{(l_1, a_1, b_1), (l_1, a_1, b_2), \dots, (l_1, a_2, b_1), (l_1, a_2, b_2), \dots, (l_n, a_n, b_{n-1}), (l_n, a_n, b_n)\}$ . Consequently, in set  $C$ , there are  $n^3$  candidates which are a color point in three-dimensional color space.



Fig. 1. An example of histograms of color components and corresponding bin center points on each histogram image.

Heuristic value of each candidate  $i$ ,  $\eta_i$ , is also determined as  $\eta_i = \gamma_i / N$ , where  $\gamma_i$  is the number of pixels which are closest to the candidate cluster center point, and  $N$  is the number of pixels in image. Thanks to determining heuristic value by this way, selecting probability of cluster centers which are more closely to the image pixels is higher than other ones.

#### 3.2. Ant colony optimization method to select optimum cluster centers set

The objective of color clustering is to divide a color set into  $k$  homogenous color clusters. Image segmentation by clustering is effective, but how to determine the number of clusters is a biggest problem in color clustering [1]. We

have been a strong belief that ACO with ants having variable solution length may tackle this problem. Therefore, ACO method for color clustering is designed as follows:

- Each ant has different solution length at each iteration. Therefore each ant select totally different number of cluster centers while constructing its solution.
- Only best ant of an iteration can deposit some amounts of pheromone (according to formula 2) on the candidate states (cluster centers), contrary to in original ACO variants (Because pheromones are usually deposited on between states in original ACO variants)
- Each ant selects next suitable state by state transition rule defined in formula 1. But, selection of next candidate is independent from the current positioned state. Instead, it selects directly based on the pheromone levels on candidate states and heuristics of candidate states without using any knowledge about previously visited states.
- Fitness value of each ant “k” solution is calculated as follows:

$$f_k = \frac{\sum_{i=1}^N \sum_{m=1}^M \min(\text{dist}(p_i, c_m))}{N} + \frac{M \times \max(M) \times \ln(M)}{n} \quad (4)$$

where  $M$  is the number of cluster centers in ant solution,  $p_i$  is a pixel and  $c_m$  is a cluster center in the ant solution,  $n$  is the number of bins in a histogram and  $\max(M)$  represents the number of cluster centers which can be in a solution maximally. This fitness function assists not only selecting optimum cluster center points but also determining minimum number of suitable cluster centers for clustering.

- At the end of each iteration, if new best ant is found, solution is updated and finally best-so-far ant solution is determined as the solution of the algorithm after termination criteria is met.

### 3.3. Cluster merging and assigning pixels to each cluster

After determining cluster centers by ACO method, it can be observed that some cluster centers are too close the each others. This may lead to oversegmentation, i.e., homogeneous regions with narrow color transition may be split as separate regions, or maybe very small regions can be generated. Therefore, these cluster centers need to be combined.

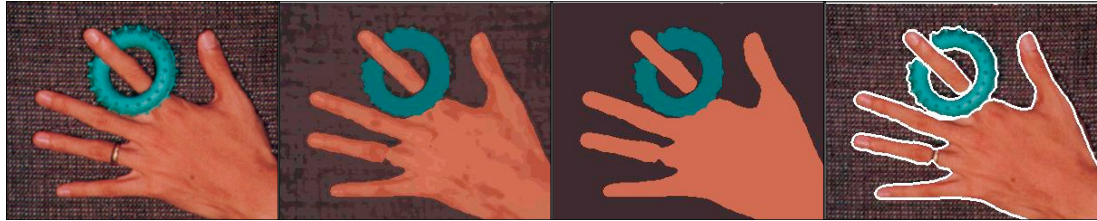


Fig. 1. Left to right: original image, segmentation result after ACO clustering, segmentation result after cluster merging, and segmented cluster regions after cluster merging

We define a simple merging criterion according to standard deviation of pixel value distribution in image. If distance between two cluster centers is smaller than the standard deviation value, then we merged the cluster center and we defined new cluster center point as follows:

$$C_{merged}(l, a, b) = \frac{\gamma_1 C_1(l, a, b) + \gamma_2 C_2(l, a, b)}{\gamma_1 + \gamma_2} \quad (5)$$

After merging, pixels are assigned to their closest cluster and image is segmented into regions. Figure 2 clearly shows a result of ACO clustering and merging effect on a test image

#### 4. Experimental Results

The proposed ACO method has been designed not only for general color image segmentation but also for a specific domain. To evince that claim, we have done experiments on two different image datasets. The first dataset consists of natural color images. Second dataset is Oxford-17 flowers dataset, for which we selected as a case study, contains flower images. All experiments have done on Matlab® with the parameter set:  $\alpha = 1$ ,  $\beta = 2$ ,  $\rho = 0.15$ , number of ants = 5, number of iterations = 20,  $n$  (number of bins) = 15, and  $\max(K) = 15$ .

##### 4.1. Results on some images of Berkeley segmentation benchmarking dataset

Berkeley dataset (<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/>) contains a large number of natural images, each of which has been hand-segmented by human observers, while the benchmark provides a methodology for quantifying segmentation performance. But, statistical and comparison results are not included in this study because of page limit. Instead, to show that the proposed method is suitable for natural image segmentation, only some visual segmentation results are shown only in Figure 3. As seen images in Figure 3, the proposed method can separate image into homogeneous regions.



Fig. 1. First column: original images, second column: segmentation results before cluster merging, third column: segmentation results after cluster merging, and forth column: segmented regions with detected boundaries.

##### 4.2. Case Study: Flower region extraction on Oxford-17 Flowers dataset

As a case study of object boundary extraction, the proposed methodology applied on Oxford-17 Flowers dataset (<http://www.robots.ox.ac.uk/~vgg/data/flowers/17/index.html>). The dataset consists of 17 flower species. But in experiments, 753 of 14 flower species have been used. Because, according to the dataset owners [14], other 4 species are not sufficient. Also, the dataset consists trimap images which are manually ground truth labeled into foreground (red pixels), background (blue pixels) and close to flower boundary regions (black pixels) for all 753

images. The regions which are shown in black pixels are difficult to classify; therefore, they have no effect on performance measurement. The performance measure  $P$  for the dataset is defined as [14]:

$$P = \frac{\text{trueforeground} \cap \text{segmentedforeground}}{\text{trueforeground} \cup \text{segmentedforeground}} \quad (6)$$

According to this measure, perfect segmentation is achieved if  $P$  is equal to 1. We have compared our method with two well-known flower segmentation methods: Saitoh et al. method [15] and Nilsback and Zisserman method [14]. According to the comparison,  $P$  score for Saitoh et al. method [15] is 0.82, and it is 0.94 for Nilsback and Zisserman method [14] on average. Our segmentation score is determined as 0.87 on average (see some results in Figure 4). Therefore, our method is better than Saitoh et al. method [15], but worse than Nilsback and Zisserman method [14]. At this point, it is important to note that median of  $P$  scores over all images reaches 0.98, this means more than 350 images of dataset our method can perfectly extract the flower regions from images. For only 50 images in which flower regions color are too close to background or flowers regions consist of distinctive colors, our proposed method is giving very bad results ( $P < 0.5$ ) yielding performance decreasing.



Fig. 1. First column: original images, second column: trimap images, third column: segmentation results after cluster merging, and forth column: final extracted flower regions.

## 5. Conclusion

In this paper, an ACO method has been proposed to tackle color image segmentation problem. It is an unsupervised method which tries to determine both number of clusters and optimum cluster centers. A simple cluster merging strategy has been also applied to tackle oversegmentation problem after ACO-based clustering. The proposed method has been tested on two different datasets. Results on images of Berkeley dataset show that the



proposed method can be used for natural color images. On the other hand, the calculated accuracy of experiments on Oxford-17 flowers dataset clearly indicate that ACO-based clustering method can tackle flower region extraction problem in color images.

It is important to note that parameters are selected arbitrary (only using observation and parameters set which are used for solving other problems) in this paper. Therefore, as a future work, we will use parameter tuning algorithms for both selecting better parameter configuration and redesigning of proposed ACO method. Thus, we will try to make algorithm more stable and improve its performance. Also, we will more focus on segmentation of natural images, and the experimental and statistical results on Berkeley dataset will be represented in detailed.

## References

1. H.D. Cheng, X.H. Jiang, Y. Sun and J. Wang, Color image segmentation: advances and prospects, *Pattern Recognit.* 34-12 (2001) 2259-2281.
2. A. Trémeau and P. Colantoni, Regions adjacency graph applied to color image segmentation, *IEEE Trans. Image Process.* 9-4 (2000) 735–744.
3. D.K. Panjwani, G. Healey, Markov random field models for unsupervised segmentation of textured color images, *IEEE Trans. Pattern Anal. Mach. Intell.* 17 (10) (1995) 939–954.
4. Y.I. Ohta, T. Kanade and T. Sakai, Color information for region segmentation, *Comput. Graph. Image Process.* 13 (1980) 222–241.
5. L. Shafarenko, M. Petrou and J.V. Kittler, Histogram based segmentation in a perceptually uniform color space, *IEEE Trans. Image Process.* 7-9 (1998) 1354–1358.
6. S.H. Park, I.D. Yun and S.U. Lee, Color image segmentation based on 3D clustering: morphological approach, *Pattern Recognit.* 31-8 (1998) 1061–1076.
7. T. Huntsberger, C. Jacobs and R. Cannon, Iterative fuzzy image segmentation, *Pattern Recognit.* 18-2 (1985) 131–138.
8. T. Uchiyama and M.A. Arbib, Color image segmentation using competitive learning, *IEEE Trans. Pattern Anal. Mach. Intell.* 16-12 (1994) 1197–1206.
9. P. Scheunders, A genetic c-means clustering algorithm applied to color image quantization, *Pattern Recognit.* 30-6 (1997) 859–866.
10. M. Dorigo, M. Birattari and T. Stützle, Ant colony optimization: artificial ants as a computational intelligence technique, *IEEE Comput. Intell. Mag.* (1999) 28–38.
11. R. Arnay, L. Acosta, M. Sigut and J. Toledo, Ant colony optimization algorithm for detection and tracking of non-structured roads. *Electron. Lett.* 44-12 (2008) 725-727.
12. M.G. Cinsdikici and D. Aydın, Detection of blood vessels in ophthalmoscope images using MF/ant (matched filter/ant colony) algorithm, *Comp. Meth. and Prog. in Biomed.* 96:85-95.
13. Z. Yu, O. C. Au, R. Zou, W. Yu and J. Tian, An adaptive unsupervised approach toward pixel clustering and color image segmentation, *Pattern Recognit.* 43 (5) (2010) 1889-1906.
14. T. Saitoh, K. Aoki and T. Kaneko, Automatic recognition of blooming flowers. In *Proc. of Int. Conf. on Pattern Rec.* (2004) 27-30.
15. M.-E. Nilsback and A. Zisserman, Delving into the whorl of flower segmentation. In *Proc. of the British Mach. Vis. Conf.* (2007) 570-579.