Pattern Recognition 43 (2010) 1889–1906

Contents lists available at ScienceDirect

Pattern Recognition

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

An adaptive unsupervised approach toward pixel clustering and color image segmentation ☆

Zhiding Yu [a,*], Oscar C. Au [a], Ruobing Zou [b], Weiyu Yu [b], Jing Tian [b]

[a] Department of Electronic & Computer Engineering, the Hong Kong University of Science & Technology, Clear Water Bay, Kowloon, Hong Kong SAR, PR China
[b] School of Electronic & Information Engineering, South China University of Technology, Guangzhou, 510641, PR China

ARTICLE INFO

Article history:
Received 24 November 2008
Received in revised form
24 October 2009
Accepted 14 November 2009

Keywords:
Ant system
Clustering
Fuzzy C-means
Image segmentation

ABSTRACT

This paper proposes an adaptive unsupervised scheme that could find diverse applications in pattern recognition as well as in computer vision, particularly in color image segmentation. The algorithm, named Ant Colony–Fuzzy C-means Hybrid Algorithm (AFHA), adaptively clusters image pixels viewed as three dimensional data pieces in the RGB color space. The Ant System (AS) algorithm is applied for intelligent initialization of cluster centroids, which endows clustering with adaptivity. Considering algorithmic efficiency, an ant subsampling step is performed to reduce computational complexity while keeping the clustering performance close to original one. Experimental results have demonstrated AFHA clustering's advantage of smaller distortion and more balanced cluster centroid distribution over FCM with random and uniform initialization. Quantitative comparisons with the X-means algorithm also show that AFHA makes a better pre-segmentation scheme over X-means. We further extend its application to natural image segmentation, taking into account the spatial information and conducting merging steps in the image space. Extensive tests were taken to examine the performance of the proposed scheme. Results indicate that compared with classical segmentation algorithms such as mean shift and normalized cut, our method could generate reasonably good or better image partitioning, which illustrates the method's practical value.

1. Introduction

Image segmentation refers to the process in which spatially connected pixels that share certain visual characteristics are assigned with the same label. The major goal is to partition an input image into multiple segments so that objects and boundaries could be located and the image could become more meaningful and easier to analyze [43]. In image processing [1–7] and computer vision [8–17], segmentation has long been considered one of the most important problems since it plays an indispensable preliminary processing role for semantic analysis and many other advanced tasks [18–24]. Indeed, accurately segmenting out objects or regions that appeal to the human vision is a significant issue. Though much emphasis has been put on this topic and many approaches have been proposed, it is still challenging to segment natural images due to their inherent complexity [25]. The simplest case might be the segmentation of an image with distinct and homogeneous foreground objects. In this case, techniques such as thresholding and edge detection using gradient information could be sufficient, even though they are relatively simple. Recent researches have also concentrated on applying thresholding with intelligent algorithms like ant colony optimization (ACO) and fuzzy measures [26,27] so that more adaptive and accurate decisions could be made to achieve better results. Malisia and Tizhoosh [26] proposed an image binary thresholding method in which ant pheromone information is added to the original image data using an ACO-based approach. The whole dataset is then clustered with the K-means algorithm. This is essentially a dimension-increasing method in the feature space and only works well with grayscale images containing distinct foreground objects. Han and Shi [28] proposed a fuzzy Ant System pixel clustering method for image segmentation. In their paper, three kinds of features including the grayscale value, the gradient value and the pixel neighborhood information are extracted to form a 3-D vector for each pixel. They define each ant's membership degree to a clustering center as the probability of going in that direction, calculated according to the AS algorithm. The segmentation performance depends largely on the image feature extraction method and the algorithm does not especially seek for very compact clustering results in the feature space. The convergence process is very fast and only one round of

☆This work has been supported by the Key Program of National Natural Science Foundation of China (Grant no. U0835001) and supported in part by the Research Grants Council (RGC) of the Hong Kong Special Administrative Region, China. (Project no. 610109).

* Corresponding author.
E-mail address: hsfzchrisding@hotmail.com (Z. Yu).

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.11.015

iteration is performed, according to [53]. The initial clustering centers are manually selected by inspecting the image grayscale histogram, which could be troublesome since some images might not contain significant histogram peaks.

The above two methods assume that the input images mostly contain uniformly colored objects, which is typically not true for natural color images. The existence of object texture, noise and shading all contribute to the variation of object colors. Given limited, low level pixel-wise features, one commonly finds it hard to directly extract foreground objects with high accuracy. Recently, there is a widely held opinion which describes the partitioning of an image as an inherently hierarchical process. In other words, it is more appropriate to think of returning a tree structure corresponding to a hierarchical partition instead of a single "flat" partition [15]. Some typical examples include the family of region merging based methods in which segmentation algorithms that were originally proposed to deal directly with natural color images have also served preprocessing roles. For example, mean shift (MS) is a well known non-parametric feature space analysis algorithm that searches for local maximal density points and then groups all the data into the clusters defined by these maximal density points. Comaniciu and Meer [29] further investigated this algorithm and implemented it in color image segmentation. Since then, MS has been well known for its good segmentation performance. This approach was also utilized by Tao, Jin and Zhang [30] to first over-segment a natural image instead of directly generating a segmentation result, to be followed by a region merging process with normalized cut (Ncut). This could achieve more accurate segmentation results than pixel-wise Ncut. Many other region merging based methods applied watershed algorithm as an image simplification operation [31,32], with a tendency to generate over-segmented results.

The concept behind this hierarchical segmentation strategy is quite straightforward: a pre-segmented region allows many more features to be defined for image analysis than a mere single image pixel. In addition to the mean vector, other such region-based features include boundary smoothness, boundary length, boundary gradient, region pixel number, region geometry, spatial topological relationships among regions. The combination of these features allows more versatile and powerful methods to be developed to achieve better segmentation. In addition, people find that fewer inputs sometimes could bring better and more accurate results using the same algorithm [30]. Finally, region merging based methods tend to require significantly lower computational complexity than pixel-wise methods, since an image often contains much fewer regions than pixels. Shi stated in his paper that, with region number much smaller than total pixel number [15], exhaustive search in the discrete domain for cost function minimization became feasible.

A relevant issue that should not be over looked is that the pre-segmentation stage, viewed as an image simplification step, should be as precise as possible. Despite such operation's task-oriented nature, people under many circumstances would want to minimize false classification while optimizing region partitioning and uniformity. Although one can partition an image into very fine regions so that the chance of false classification is small, such strategy could bring extra computation complexity to the subsequent operations and risk losing local information that might be otherwise helpful. Based on different objectives and needs, people have designed various image pre-segmentation methods. The idea of superpixel was originally developed by Ren and Malik using normalized cut to learn a classification model for segmentation [44], whereas MS and watershed algorithm could also serve similar preprocessing roles. Other than these, FCM has long been a popular algorithm in computer vision and pattern recognition for its clustering validity and simplicity of implementation. FCM-based pixel clustering tends to group pixels together according to similarities in the feature space. Given fixed number and distribution of clusters, segmented regions are decided depending on the pixel similarity and spatial connectivity of the image. Unlike superpixel, FCM does not have very strong constraint on the segmented region area, meaning that if an image contains large homogeneous area, it is likely to be segmented out as a whole. Compared with MS, FCM tends to preserve more details along segmented region boundaries, especially texture boundaries, which could be helpful for further texture analysis. Although FCM is an excellent image preprocessing tool, its implementation often faces two unavoidable initialization problems:

1. How should we decide the cluster number?
2. How should the initial centroids be properly distributed?

These problems have their unneglectable impacts on the segmentation quality. While the first issue could largely affect segment area and region tolerance for feature variance, the second one affects the cluster compactness and classification accuracy. In this paper, we propose a novel ACO-based scheme called AFHA to address these two problems. With the help of AS, we achieve adaptive cluster initialization and improved clustering structure. To extend its application to image segmentation, we also consider the spatial connectivity issue of pixel clustering and propose relevant solutions. We will show that, even with simple postprocessing, our method could generate reasonable segmentation results.

The rest of the paper is organized as follows. Section 2 will summarize previous works that are most related to our research. Section 3 will describe our proposed AFHA while Section 4 will give experimental results and both qualitative and quantitative evaluation. Finally, conclusions will be made in Section 5.

2. Background

2.1. The Fuzzy C-means algorithm

Data clustering is the process of dividing data elements into classes or clusters so that elements in the same class are as similar as possible, and those in different classes are as dissimilar as possible. In fuzzy clustering, data elements can belong to more than one cluster, and associated with each element is a set of membership levels. These indicate the strength of the association between that data element and a particular cluster. One of the most widely used fuzzy clustering algorithms is the Fuzzy C-means. Introduced by Ruspini [34] and improved by Dune and Bezdek [35,36], the FCM algorithm attempts to partition a finite dataset $\boldsymbol{X}=\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N\}$ into a collection of M fuzzy clusters with respect to some given criterion [37], which is essentially a Hill-Climbing technique. Let m be the exponential weigh of membership degree, $m\in[1,\infty)$. The objective function W_m of FCM is defined as:

$$W_m(\boldsymbol{U},\boldsymbol{C})=\sum_{i=1}^{N}\sum_{j=1}^{M}(\mu_{ji})^m(d_{ji})^2, \tag{1}$$

where μ_{ji} is the membership degree of $\boldsymbol{x}_i$ to $\boldsymbol{c}_j$ and d_{ji} is the distance between $\boldsymbol{x}_i$ and $\boldsymbol{c}_j$. Let $\boldsymbol{U}_i=(\mu_{1i},\mu_{2i},\ldots,\mu_{Mi})^{\mathrm{T}}$. Then $\boldsymbol{U}=(\boldsymbol{U}_1,\boldsymbol{U}_2,\ldots,\boldsymbol{U}_N)$ is the membership degree matrix and $\boldsymbol{C}=\{\boldsymbol{c}_1,\boldsymbol{c}_2,\ldots,\boldsymbol{c}_M\}$ is the set of cluster centroids. W_m indicates the compactness and uniformity degree of clusters. Generally, a smaller W_m reflects a more compact cluster set.

Since there is no close form solution, the minimization of W_m is an iteration process mathematically described as follows:

(1) Initialize m, M and initial cluster centroid set $\boldsymbol{C}^{(0)}$. Set the iteration terminating threshold ε to a small positive value and iteration time q to zero. Calculate $\boldsymbol{U}^{(0)}$ according to $\boldsymbol{C}^{(0)}$ with the following equation:

$$\mu_{ji} = 1/\sum_{k=1}^{M}(d_{ji}/d_{ki})^{2/(m-1)} \quad J_i = \phi \tag{2}$$

where $J_i = \{j | 1 \le j \le M, d_{ij} = 0\}$, $\sum_{j \in J_i} \mu_{ji} = 1$. Notice that if $d_{ji} = 0$, then $\mu_{ji} = 1$ and set other membership degrees of this pixel to zero.

(2) Calculate $\boldsymbol{C}^{(q+1)}$ according to $\boldsymbol{U}^{(q)}$ with the following equation:

$$\boldsymbol{c}_j = \frac{\sum_{i=1}^{N}(u_{ji})^m \boldsymbol{x}_i}{\sum_{i=1}^{N}(u_{ji})^m} \tag{3}$$

(3) Calculate $\boldsymbol{U}^{(q+1)}$ according to $\boldsymbol{C}^{(q+1)}$ with Eq. (2).

(4) Compare $\boldsymbol{U}^{(q+1)}$ with $\boldsymbol{U}^{(q)}$. If $\|\boldsymbol{U}^{(q+1)} - \boldsymbol{U}^{(q)}\| \le \varepsilon$, stop iteration. Otherwise, go to (2).

2.2. The ant system

Inspired by ant behaviors, Marco Dorigo et al. in 1991 proposed the first ant colony algorithm (ACA), Ant System, and successfully applied it in optimizing the solutions of traveling salesman problem [38]. AS exhibited many advantages in solving discrete optimization problems. In addition it afterward inspired a variety of improved ant colony algorithms [39]. Path construction and pheromone update are two major steps of AS. In the path construction step, let path (i, j) denote the path which connects node i and j. When going from i to j, an ant chooses its path with the following probability:

$$p_{ij} = \frac{ph_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{s \in S} ph_{is}^{\alpha}(t)\eta_{is}^{\beta}(t)} \quad j \in S, \tag{4}$$

where $\eta_{ij}(t) = 1/d_{ij}$ stands for the heuristic information and $d_{ij} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$ denotes the distance between i and j. The $\boldsymbol{x}_i$ can be a value or a vector that is characterized by node i. Function $ph_{ij}(t)$ represents the pheromone concentration on path (i, j) at time t. The S is the set of all available paths.

In the pheromone update step, pheromone concentration on every path is updated according to the following equation:

$$ph_{ij}(t') = \rho ph_{ij}(t) + \Delta ph_{ij}, \tag{5}$$

where ρ represents the evaporating degree of pheromone concentration with the elapse of time. Δph_{ij} is the increase of pheromone concentration on path (i, j) after one cycle:

$$\Delta ph_{ij} = \sum_{k=1}^{N} \Delta ph_{ij}^{k}, \tag{6}$$

where Δph_{ij}^{k} is the pheromone concentration left on path (i, j) by the kth ant.

3. Proposed approach

3.1. The ant colony–fuzzy C-means hybrid algorithm

The iterative optimization of FCM is essentially a local searching method, which is likely to fall onto a local minimum point and is very sensitive to the initialization condition of cluster centers and centroid number. Usually, initialization is carried out based on certain experience. Clustering result depends largely on whether parameters have been properly chosen. When confronted with massive data of high dimensions, it is hard to both manually and properly set parameters without repetitive experiments, which is a laborious operation likely to generate sub-optimized image segmentation results. In this paper, we apply improved Ant System to initialize FCM in view of the drawbacks. The main underling principle is to use the robustness of ACA to overcome FCM's sensitiveness to the initialization condition. Moreover, its intelligent searching ability will help to further achieve optimization.

In order to improve its performance for clustering tasks, relevant modifications should be made for AS. AS is well known to suffer from high computational complexity, as indicated in many previous researches. When applied in clustering problems, AS tends to be time consuming because for every pixel in an image, distances and pheromone concentration on the paths that lead to all the other ants have to be calculated, requiring tremendous computation. In addition, computation requirement will be tripled if every ant is a 3-D vector instead of a 1-D one. To solve this problem, we choose M' cluster centers based on color quantization and hence every ant only needs to calculate its distances to these "food sources". Computation can be reduced because M' is usually much smaller than N, the total number of pixels. We further confine the computation by setting the cluster radius. If the distance between an ant and a cluster center is larger than a given radius, then its probability of going to that "food source" is set to zero. In this fashion, ants will simply ignore clusters that are too far away. This may further bring higher clustering accuracy and smaller computational complexity to the algorithm.

In each round of iteration, ants previously classified will no longer be considered. Only those unclassified will take part in the clustering process which is a hard probabilistic partition, different from Han and Shi's method. The algorithm will accelerate as more and more ants become classified. There are two possible convergence conditions: either all ants are classified, or there are still unclassified ants but the clustering centroids remain unchanged for a certain number of iterations.

Another concerned problem is stagnation, which comes from excessive pheromone accumulation on a single path. AS could easily get into stagnation as a result of its product form of heuristic information and pheromone concentration in the probability equation. This situation is especially likely to happen when solving clustering problem with AS, for massive number of ants may choose the same cluster center at a time, leaving intense pheromone concentration which attracts even more ants in the next round. We propose a probability equation with the summation form of heuristic information and pheromone concentration in order to improve its robustness against massive build up of pheromone. When an ant is in close proximity to a food source, heuristic information will play the dominant role and the ant will most likely be attracted to this food source. This kind of formation helps to reduce the chance that an ant goes erroroneously to a food source much further away because of huge pheromone concentration at that point.

Together with the algorithm definitions of FCM and AS in Section 2, AFHA can be described as follows:

(1) Read image data to obtain a 3-D dataset, with each element being a 3-D vector containing 3 components representing the 3 colors of an image pixel.

(2) Initialize cluster centroid set $\boldsymbol{V} = \{\boldsymbol{v}_h | \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{M'}\}$ and centroid number M' based on pixel color statistics.

- For each color channel of *RGB*, quantization is performed to reduce the cell number from 256 to 8, each cell with a width of 32. The collection of all possible quantized colors forms a feature vector set $\boldsymbol{C}'$. Let $\boldsymbol{c}'_l$ be the lth element of $\boldsymbol{C}'$. Then $\boldsymbol{c}'_l$ is a 3-D vector representing the set element, with each dimension c'^k_l being a cell endpoint. In other words, $c'^k_l = 32 \cdot n_{lk}$, where $1 \le k \le 3$, $0 \le n_{lk} \le 8$, $1 \le l \le 729$. $\boldsymbol{c}'_{l_1} \ne \boldsymbol{c}'_{l_2}$, if $l_1 \ne l_2$.
- Set up mapping between pixels and the above quantized vectors. For every pixel in the image, each dimension of its feature vector is rounded to $32n$, where n is an integer ranging from 0 to 8. This pixel is then assigned to the corresponding $\boldsymbol{c}'_l$.
- Let $\boldsymbol{X}_l$ represent the pixel set that are assigned to $\boldsymbol{c}'_l$ and $p_l = |\boldsymbol{X}_l|$ denote the number of pixels assigned to $\boldsymbol{c}'_l$. Let v_h be the mean value of the pixels assigned to $\boldsymbol{c}'_l$, whose $p_l \ge P$ for some threshold P. $\boldsymbol{V} = \{\boldsymbol{v}_h | \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{M'}\}$. For $1 \le l \le 729$, if $p_l \ge P$, calculate $\boldsymbol{v}_h$ with the following equation:

$$\boldsymbol{v}_h = \frac{1}{p_l} \sum_{\boldsymbol{x}_i \in \boldsymbol{X}_l} \boldsymbol{x}_i \tag{7}$$

where $1 \le h \le M'$, M' is the number of $\boldsymbol{c}'_l$ whose p_l is larger than P.

(3) Calculate cluster center set $\boldsymbol{C}$ and center number M using improved AS.
- Let $\boldsymbol{V}_h$ represent the pixel set that is assigned to cluster center $\boldsymbol{v}_h$. Set $q=0$, $M=0$.
- For each unclassified ant $\boldsymbol{x}_i$, calculate its distance d to every cluster center $\boldsymbol{v}_h$, where $d_{ih} = \|\boldsymbol{x}_i - \boldsymbol{v}_h\|$. Set $p_{ih}=1$ if $d_{ih}=0$, while other probabilities are set to 0.
- If $d_{ih} \ne 0$, calculate the probability according to the following equation:

$$p_{ih} = \begin{cases} \dfrac{ph_h^{\alpha}(t) + \eta_{ih}^{\beta}(t)}{\sum_{s \in S}(ph_s^{\alpha}(t) + \eta_{is}^{\beta}(t))} & h \in S \\ 0 & \text{otherwise} \end{cases}, \tag{8}$$

where $\eta_{ih}(t) = r/d_{ih}$, $S = \{s | d_{is} \le r, s = 1, 2 \ldots M'\}$.
- If the probability is larger than a given threshold λ, assign the ant to cluster center $\boldsymbol{v}_h$. Otherwise, leave this ant to the set waiting for next iteration. Update the pheromone concentration on $\boldsymbol{v}_h$ with the following equations:

$$ph_h(t') = \rho ph_h(t) + \Delta ph_h^k, \tag{9}$$

$$\Delta ph_h = \sum_{k=1}^{N} \Delta ph_h^k, \tag{10}$$

where Δph_h is the increase of pheromone concentration on $\boldsymbol{v}_h$ after one cycle, Δph_h^k is the pheromone concentration left on $\boldsymbol{v}_h$ by the k^{th} ant.
- Calculate the cluster centroids by the following equation:

$$\boldsymbol{v}_h = \frac{1}{|\boldsymbol{V}_h|} \sum_{\boldsymbol{x}_i \in \boldsymbol{V}_h} \boldsymbol{x}_i \quad \text{if } |\boldsymbol{V}_h| \ne 0 \tag{11}$$

- Calculate the distance between cluster centroids. If the minimum distance is less than a given threshold dc, we merge the two nearest clusters to form a new one. Add their pheromone concentration and refresh the center value by Eq. (11).
- $q = q+1$.
  If $Iter \ge T$, or there is no unclassified ant, stop iteration and go to the last step.
  Else,
  if $\|\boldsymbol{V}^{(q+1)} - \boldsymbol{V}^{(q)}\| \ge \varepsilon_1$, go to the second step in phase (3), $Iter=0$.
  Else, go to the second step of phase (3), $Iter = Iter+1$.
- For every cluster center, if $|\boldsymbol{V}_h| \ne 0$, $\boldsymbol{c}_j = \boldsymbol{v}_h$. $M = M+1$. $\boldsymbol{C}^{(0)} = \{\boldsymbol{c}_j | \boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_M\}$.

(4) pixel clustering based on FCM
- Set iteration terminating threshold ε_2 to a small positive number and iteration time number q to 0. Initialize $\boldsymbol{U}^{(0)}$ according to $\boldsymbol{C}^{(0)}$ with Eq. (2).
- Update $\boldsymbol{C}^{(q+1)}$ according to $\boldsymbol{U}^{(q)}$ with Eq. (3).
- Update $\boldsymbol{U}^{(q+1)}$ according to $\boldsymbol{C}^{(q+1)}$ with Eq. (2).
- Compare $\boldsymbol{U}^{(q+1)}$ with $\boldsymbol{U}^{(q)}$.
  If $\|\boldsymbol{U}^{(q+1)} - \boldsymbol{U}^{(q)}\| \le \varepsilon_2$, stop iteration.
  Otherwise, $q = q+1$. Go to the second step of phase (4).

3.2. Illustration of the implementation procedure

To illustrate the implementation process, we apply this proposed approach to perform pixel clustering with the 256×256 image *Peppers* depicted in Fig. 1(a). The clustered results of different stage are respectively, illustrated in Fig. 1(b–d). Experiments were taken under the environment of *Matlab R2007b*, with an *Intel Core2 Duo P8600* CPU (2.4 GHz). First, image data is loaded and initial cluster center set $\boldsymbol{V}$ is decided according to phase (2) described in Section 3.1. The centroids after color quantization are depicted by blue crosses points in Fig. 2. Next, $\boldsymbol{C}^{(0)}$is obtained with improved AS, which is mathematically described by phase (3). The result of $\boldsymbol{C}^{(0)}$ is illustrated by red crosses in Fig. 2. Notice there is an obvious reduction of cluster number since improved AS includes a cluster merging step, an operation that automatically keeps a reasonable cluster number for all kinds of input images. Finally, with initialized cluster centers and center number indicated by $\boldsymbol{C}^{(0)}$, image pixels are clustered using FCM algorithm described by phase (4). Final cluster centers are depicted in Fig. 2 by green squares.

3.3. Improving algorithm efficiency with ant subsampling

Although time is not a primary factor for optimization in precision-oriented image segmentation, it yet has important influence in algorithm practicability. Even with improvement of AS based on setting cluster centers, computational complexity of

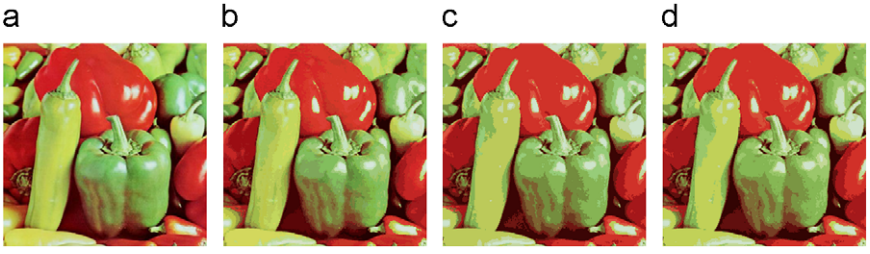

Fig. 1. Image *Peppers* and its segmentation results. (a) Original image *Peppers*. (b) Image after color quantization. (c) Image after running AS. (d) Final clustering result.

the proposed method is still much larger than usual FCM since AS is a more complicated algorithm compared with FCM. We propose an ant subsampling based method to modify AFHA. Mathematically, the procedure can be described by replacing the third step of phase (2) in Section 3.2 with the following steps:

Let $\boldsymbol{X}_l$ represent the pixel set that are assigned to $\boldsymbol{c}_l'$ and $p_l = |\boldsymbol{X}_l|$ denote the number of pixels set to $\boldsymbol{c}_l'$. $\boldsymbol{v}_h$ stands for the average value of pixels assigned to $\boldsymbol{c}_l'$, whose $p_l \geq$ threshold P. $\boldsymbol{V} = \{\boldsymbol{v}_h | \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{M'}\}$. *Per* stands for the percentage of ant number compared with N. For $1 \leq l \leq 729$, randomly select $Per \cdot |\boldsymbol{X}_l|$ pixels from $\boldsymbol{X}_l$ to form pixel set $\boldsymbol{X}_l'$ and calculate average value v_h with the following equation:

$$\boldsymbol{v}_h = \frac{1}{|\boldsymbol{X}_l'|} \sum_{\boldsymbol{x}_i \in \boldsymbol{X}_l'} \boldsymbol{x}_i, \quad \text{if } |\boldsymbol{X}_l'| \neq 0 \tag{12}$$

where $1 \leq h \leq M'$, M' is the number of $\boldsymbol{c}_l'$ whose p_l is larger than P.

In addition, at phase (3) in Section 3.2, only mark the selected $Per \cdot |\boldsymbol{X}_l|$ pixels in the above step as unclassified ants waiting for distribution instead of marking all pixels as unclassified ants.

The computational complexity of improved AFHA (IAFHA) could be largely reduced since the total ant number only takes a small proportion of the total pixel number N. Moreover, compared with original algorithm, IAFHA is able to generate a similar initialization for FCM because ants are selected with a fixed ratio from every pixel set that belongs to $\boldsymbol{c}_l'$. Notice that the smaller the proportion, the faster the improved AFHA will be. However, if *Per* is set to be too small, the improved AFHA will become an FCM algorithm with randomly selected initial cluster centers, which tends to be unstable and unreliable. Thus a tradeoff between running time reduction and keeping a good initialization is needed when choosing *Per*.

To illustrate its performance, we apply IAFHA to segment image *Peppers*, with the segmentation results and the running time comparison respectively, demonstrated in Figs. 3 and 4. In Fig. 3(b), again blue crosses represent $\boldsymbol{v}_h$ while red crosses and green squares respectively, stand for $\boldsymbol{C}^{(0)}$ and the final cluster centers. In can be inferred from Fig. 4 that compared with AFHA, there is a large running time reduction by IAFHA, which suggests a significant decrease of computational requirement.

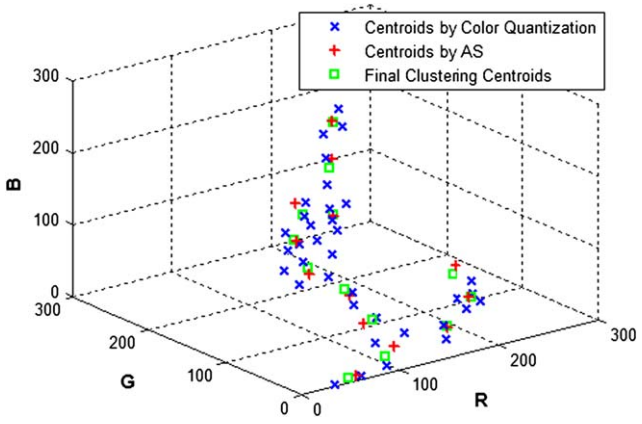

Fig. 2. Illustration of cluster centroids.

Fig. 3. Illustrations of segmentation result and cluster centers using IAFHA. (a) Image after color quantization. (b) Image after running AS. (c) Final clustering result.

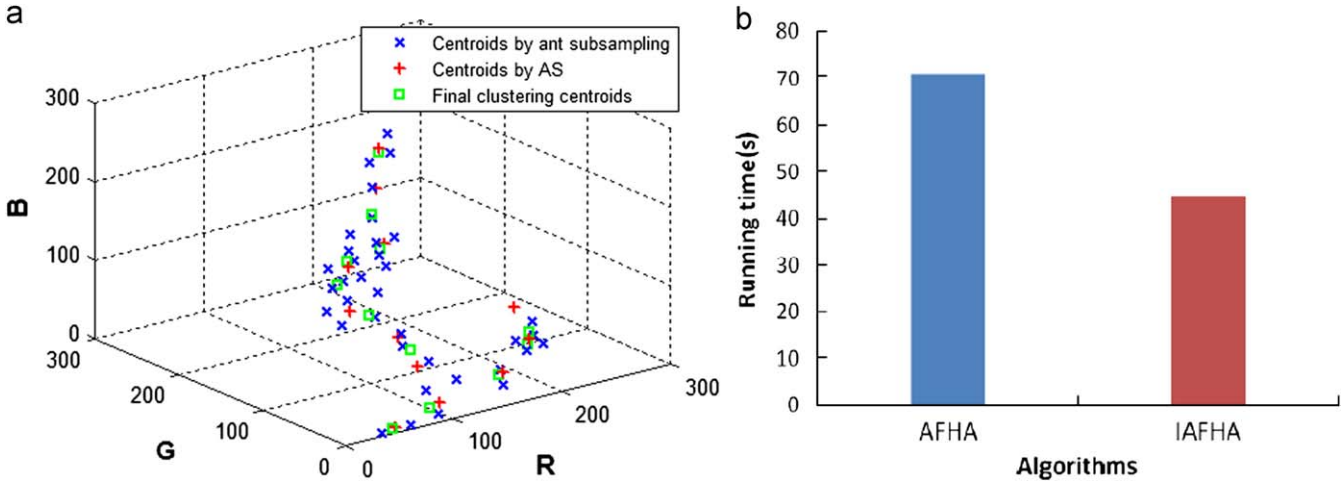

Fig. 4. (a) Illustration of cluster centroids. (b) Comparison of algorithm efficiency.

3.4. Generating spatially continuous segments

Segmentations with clustering are often featured with numerous discrete small regions. The spatial connectivity between pixels in the same cluster could hardly be guaranteed. These minor regions on one hand preserves the image detail but on the other hand largely affects the segmentation quality. To generate reasonable segmentations, postprocessing is necessary for this issue.

Some previous works have concentrated on add clustering with spatial constraints [50]. The original clustering cost function is added with a neighboring information term to enhance the spatial continuity of clustering results. They actually provide feasible ways for generating results with less noise effects and could be potentially incorporated into our proposed scheme. However, spatial connectivity is still not guaranteed since a definite spatial connectivity constraint has not been imposed. Rather, these methods are more like smoothing operations that average out noise points, although not directly but in the form of membership modification.

Instead of the above strategy, we build a region adjacency graph for the clustered image. Eight-connection is applied to define spatial connectivity and spatially connected pixels in the same cluster are assigned with the same labels. We simply set a threshold for the minimum region pixel number to eliminate small regions and greedily merge two regions together according to their region dissimilarities. The stopping criterion would either be a threshold for the region dissimilarity, or a minimum region number. The dissimilarity measure could be elaborately designed to achieve better results, but for simplicity only the Euclidean distance between the two region mean feature vectors is considered in this paper. We shall observe that, with these relatively simple operations, reasonable segmentations could be generated.

4. Experimental results

4.1. Parameter analysis

According to algorithm description, α, β, λ, ρ, dc and P are the major parameters of AFHA. α and β are two parameters that control the relative weight of the pheromone concentration and the heuristic value. Adjusting only one of them will be enough for finding a balance between heuristic information and pheromone. λ indicates the minimum probability for pixel classification. ρ stands for the evaporation rate of pheromone concentration. dc denotes the minimum distance for merging clusters. Finally, P denotes the minimum number of pixels to form an initial cluster center. For IAFHA, Per, the parameter which controls the ant proportion, is also a major parameter that controls the segmentation result.

Relevant researches and experiments have revealed some basic properties of the parameters. If α is reduced to zero, AS is essentially a greedy randomized search algorithm. However when α is assigned with a value which is too large, pheromone will take the major influence in AS and the whole algorithm will become a much less optimized one. For λ, setting it to be too small is likely to induce false classifications. On the contrary, if λ is larger than 0.5, the number of all available paths will be no larger than 1. Moreover, λ with a value too large will prevent many pixels from being classified and cause running time to be much longer. ρ should be assigned to less than 1 in order to prevent stagnation. dc should not be initialized too large or too small since it may cause over segmentation or imprecise segmentation. For P, initializing it to be too small will increase computational complexity since the initial cluster center number is greatly increased. However, if P is too large then the initialization will also be a sub-optimized one. Finally, Per should be assigned a value larger than a real positive threshold. Consider an extreme case in which Per equals to $1/|\boldsymbol{X}_l|_{\max}$. From 3.3 one immediately knows that only 1 subsampled point will be selected from the largest set, provided that $Per \cdot |\boldsymbol{X}_l| < 1$ indicates subsampling 0 point from the lth set. If Per is further increased, one can possibly get M or more subsampled points, which satisfies the basic requirement of initialization. In this case, IAFHA partially degenerates to FCM with random initialization—not totally because initialization is biased towards larger sets. From the subsampling point of view, increasing Per will generally enhance the expectation of better predicting the original dataset. In real applications, a threshold of 0.1 or larger is usually favored. An example is the subsampling ratio in Matlab function "K-means" with option "start" selected as "cluster", which adopts 0.1 as the threshold.

An important property allows us to implement this algorithm without much concern about parameter tuning. More specifically, these parameters are normally independent with each other and relatively stable for various inputs. Although one could hardly guarantee that the most optimized results will be achieved with fixed parameters, the algorithm is likely to generate reasonable and satisfactory results if the parameters are within a proper range. The reason behind is that the AS-based cluster initialization step has been designed to be robust against input and parameter changes. Color quantization enables directed and extensive search in the feature space, while cluster merging endows adaptivity of cluster numbers. Moreover, the merging process is taken in a greedy and gradual way, which further increases clustering accuracy and robustness. In general, parameters are fixed through training and the set of trained parameters could give good performance and work independently with the input. In the training process, one can inspect the reasonable range of a single parameter at a time by tuning it while fixing all other parameters. Then a parameter value is chosen and the same process is iteratively carried out for the whole parameter set under different inputs. Although trainings based on various inputs could indicate diverse "locally optimized" parameter set, the reasonable parameter ranges one observes tend to overlap and thus decisions could be made to finally determine the parameter values. Accordingly, we fix the parameters as follows: α=0.5, β=2, λ=0.4, ρ=0.8, Per=0.3, m=2, r=80. The following experiments will all adopt this parameter setting and we shall see that it could work well with different inputs. The only parameter that is dependent on the input is P, influenced by the input image size. Setting P as $0.006 \sim 0.008\,N$ will be able to generate enough initial centers for extensive and directed search in the feature space. We are particularly interested in the selection of dc since this parameter is the key parameter that influences the final cluster number and the algorithm tolerance to clustering distortion. Thus we vary this parameter in the following tests to find a reasonable value and to decrease the randomness of the clustering results.

4.2. Evaluation on clustering structures

Previous works on fuzzy clustering have featured several important cluster validity criterions for evaluation of the cluster quality. One of the most fundamental benchmark is the mean squared error (MSE), which could be described as follows:

$$MSE = \frac{1}{N}\sum_{j=1}^{M}\sum_{i \in S_j}\|\boldsymbol{x}_i - \boldsymbol{c}_j\|^2 \tag{13}$$

The concept of *MSE* is quite clear: when cluster number is fixed, a good clustering algorithm should always generate results with small distortion. In other words, cluster centroids should be placed in such a way that they reduce the distances to data pieces as much as possible. This concept can be applied to many clustering algorithms in which objective functions are expressed in terms of weighted distortions.

Another commonly used benchmark is Bezdek's partition coefficient [45], whose evaluation function *VPC* is defined as follows:

$$V_{PC} = \sum_{i=1}^{N}\sum_{j=1}^{M} \mu_{ji}^2/N \tag{14}$$

Properties of this cluster validity evaluation model were studied in [45,46]. For a crisp partition, *VPC* achieves maximum value of 1. This equation essentially measures the fuzziness of a clustering result. A smaller *VPC* value indicates a fuzzier result. From the context of validation, if an algorithm produces a result that is quite fuzzy, then this algorithm is not doing a good job. Thus the larger the *VPC* value, the better the clustering result.

A more recent validity evaluation model is the Xie–Beni function [47]:

$$V_{XB} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} \mu_{ji}^2 \|\boldsymbol{x}_i - \boldsymbol{c}_j\|^2}{N \min_{\forall j \neq k}\{\|\boldsymbol{c}_j - \boldsymbol{c}_k\|^2\}} \tag{15}$$

According to Xie and Beni, *VXB* should decrease monotonically when M is close to N. When *VXB* shows a smaller value, the result is presumably a better partition. This also accords with our segmentation applications since given certain number of cluster, we would like regions be visually as different as possible.

Both AFHA and IAFHA are implemented with parameters set in Section 4.1 to perform pixel clustering on six different color images. Moreover, tests on improved AS, randomly initialized FCM are made for comparison. Random initializations have been shown to be the best approach for the C-means family [51], and we are interested to see whether the proposed scheme could result in generally better partitions than random initialization. AS also adopts the above optimized parameters since its clustering process is a similar one to the initialization process of AFHA. To give a fair comparison, m in FCM is set to 2 and M is decided by referring to the final segment number of IAFHA. However, it should be pointed out that FCM itself does not possess an adaptive decision mechanism of M. To implement FCM, a laborious process of selecting M is generally needed.

4.2.1. MSE test

Fig. 5 shows the *MSE* values of clustering by varying *dc* from 20 to 40. Notice that both AFHA and IAFHA shows their advantages over randomly initialized FCM with less distortion and FCM has larger fluctuation with its performance, which shows the improvements of compactness and stability clustering using the proposed approach.

4.2.2. VPC Test

Fig. 6 shows the *VPC* values of clustering by varying *dc* from 20 to 40. It is interesting to see AS sometimes produces the best general distribution, while *VPC* after the distortion minimization step in AFHA and IAFHA might to some extent degenerate (See Fig. 6 (e)). However, the performance of AFHA and IAFHA is still obviously better than randomly initialized FCM.

4.2.3. VXB Test

Fig. 7 shows the *VXB* values of clustering by varying *dc*. Again, we could see results similar to *VPC*. Both AFHA and IAFHA have better cluster distributions compared with randomly initialized FCM.

4.2.4. Comparison with standard initializations

There exist some standard methods for clustering initialization in the machine learning literature. The Matlab toolbox provides K-means with multiple initialization options, including "sample", "uniform" and "cluster". The first option selects M observations from the input data at random and is the default initialization method. The second option selects M points uniformly at random from the range of the input. The third method performs a preliminary clustering phase on a random 10% subsample of input data. Notice that none of these options includes automatic decision of cluster number. Again, we always set M equal to the final cluster number of IAFHA. Tables 1–3 show the comparison of clustering performance between the proposed IAFHA and the latter two methods mentioned above. Results in the tables indicate the advantage of IAFHA over the standard initializations. Notice that the name "*S.S.C.*" is short for "*Seven_Sisters_Cliffs*".

4.3. Evaluation on pre-segmentation results

Fig. 8 shows the original images of *Car* (430×256), *Peppers* (341×256), *House* (256×256), *Football* (256×256) and *Golden Gate* (256×256) and their pixel clustering results. Notice for *Car*, AFHA, IAFHA and AS outperform the other methods by classifying the meadow and diaphragm wall as single clusters while FCM falsely clusters the diaphragm wall as part of the road. In the clustering test of *Football*, there are considerable pixels mistakenly assigned to the background by FCM, which is an obvious classification error that makes the football to be a mis-shapen one. In *Seven Sisters Cliffs*, AFHA and IAFHA also outperform FCM by accurately assigning the pixels of the house roofs and the bushes (Fig. 8).

When segmenting relatively simple natural images, AFHA and IAFHA automatically assign fewer clusters with fewer regions, which are observed in tests on *House* and *Football*. The reason lies in the mechanism that cluster center initialization is an adaptive process which decides cluster centers and center number. This will help to find a better classification and the advantage of less distortion would also be considerable when postprocessing color images. Based on experimental results, we fix *dc* as 28. Although this value would not guarantee that the best cluster number is obtained, it tends to produce reasonable results that are suitable for postprocessing (Figs. 10 and 11)

4.4. Comparison of algorithm efficiency

Though algorithm efficiency might not be our primary concern for image segmentation, running time imposes large influence upon the practicability of the proposed approach. Essentially, AFHA is an algorithm which concentrates on its performance by sacrificing its efficiency. Given M cluster numbers, the complexity of AS is approximately $O(MN)$, which is the same as FCM. Consider that AS is an algorithm that has more complicated operations than FCM, the computational complexity of AFHA is thus higher. A better solution is to find a tradeoff between them, which results in the IAFHA based on ant subsampling. The advantage of IAFHA lies in the fact that computational complexity is reduced while keeping its performance close to AFHA, which is proven by clustering structure tests in Sections 4.2 and 4.3. Table 4 shows the execution time of the four algorithms when dc=28, while Fig. 9 gives the visualized contrast of efficiency.

4.5. Comparison with the X-means

There are also some state-of-the-art techniques which aim at providing a solution to the initialization problem, especially the decision of the cluster number. The X-means algorithm is proposed for the purpose of automatic decision of cluster number using the Bayesian information criterion (BIC) [52]. The algorithm splits each parent cluster into two sub-clusters by performing a local K-means

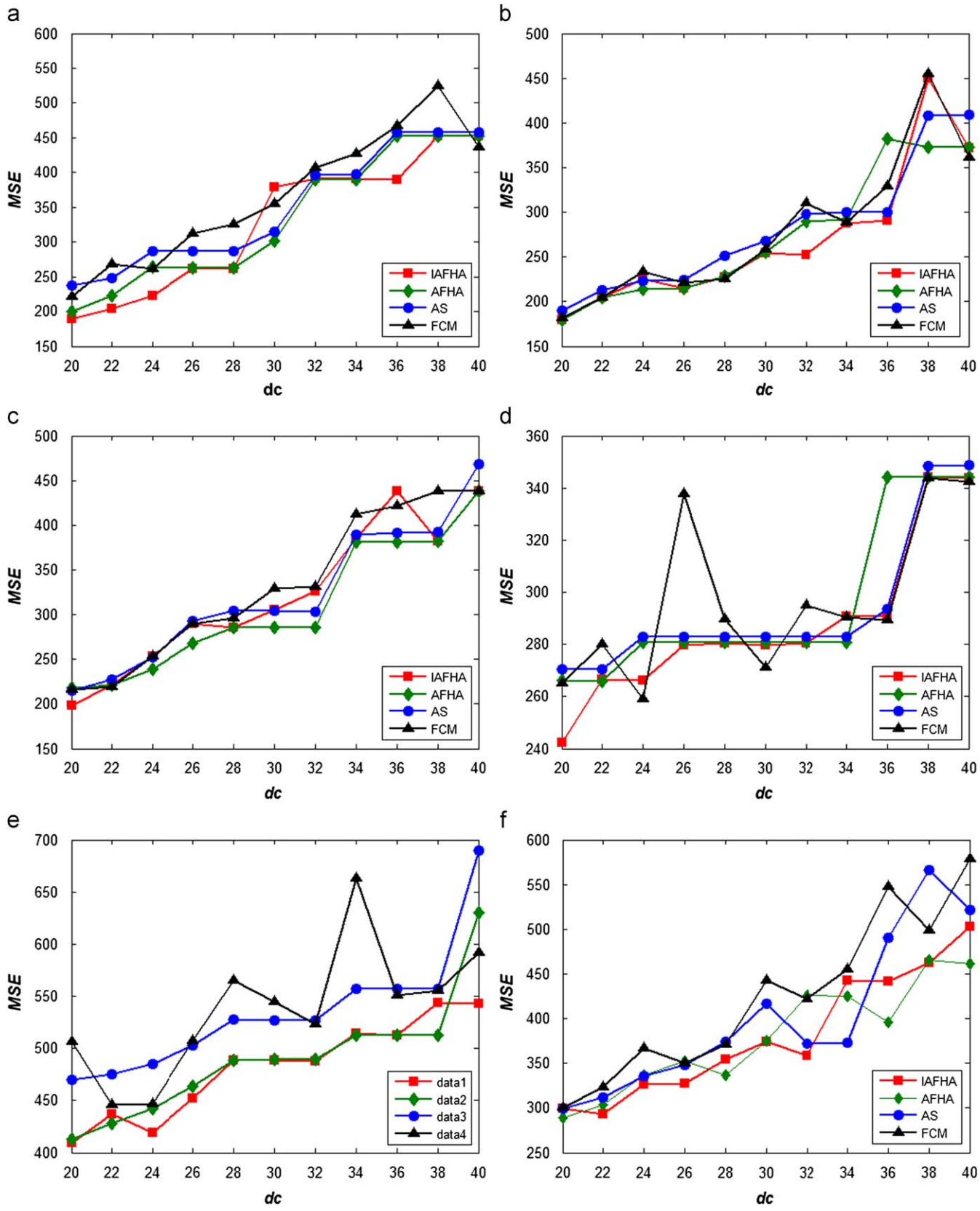

Fig. 5. Illustration of *MSE*. (a) *Car*. (b) *Football*. (c) *Golden Gate*. (d) *House*. (e) *Peppers2*. (f) *Seven Sisters Cliffs*.

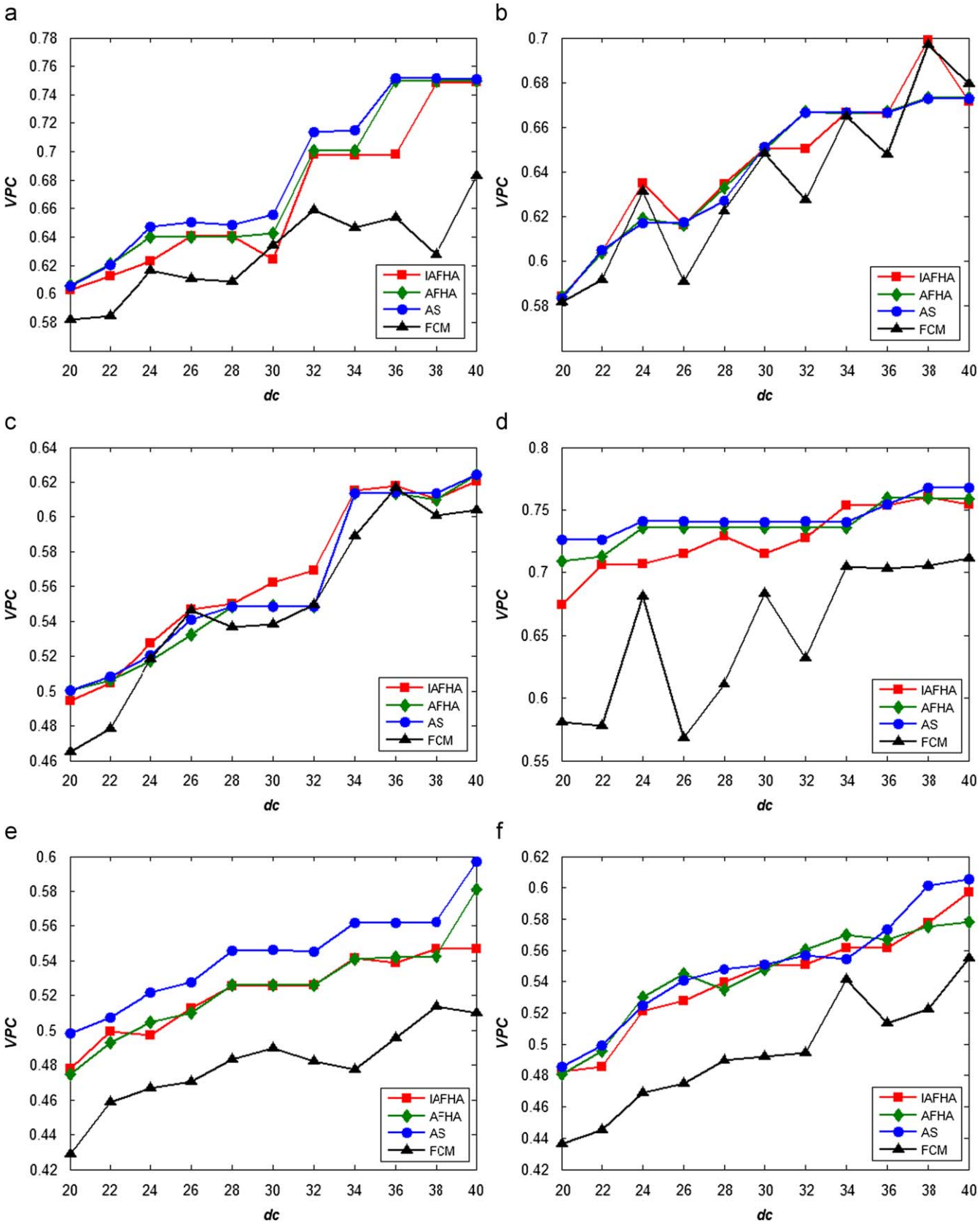

Fig. 6. Illustration of *VPC*. (a) *Car*. (b) *Football*. (c) *Golden Gate*. (d) *House*. (e) *Peppers2*. (f) *Seven Sisters Cliffs*.

clustering. Based on local BIC values before and after cluster splitting, the algorithm decides whether to keep the parent cluster or to split it into two. Usually, initial cluster number is set to be very small. Global clustering and local cluster splitting is iteratively performed to increase the cluster number until the maximum number is reached. Global BIC is also calculated to select the best model encountered.

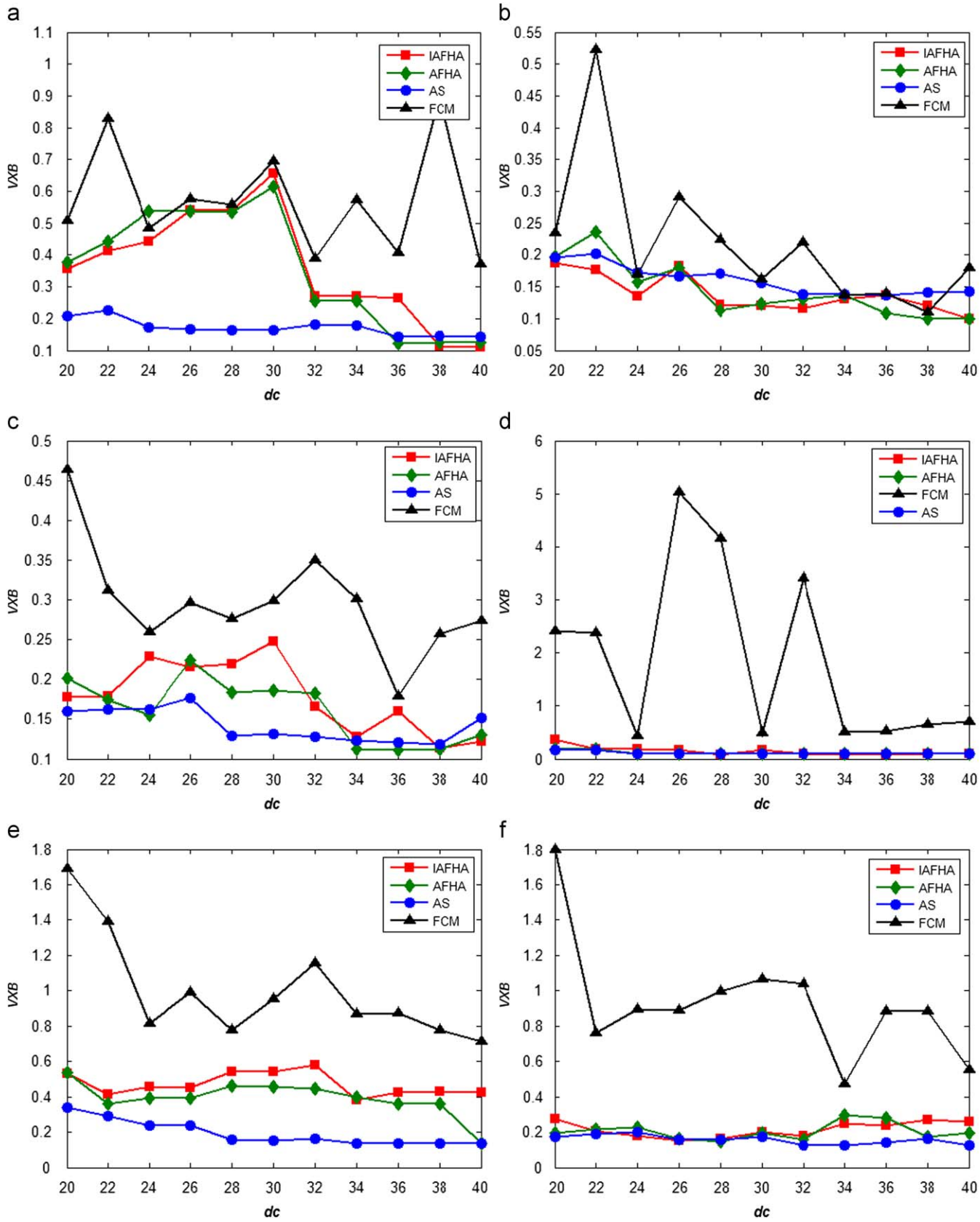

Fig. 7. Illustration of *VXB*. (a) *Car*. (b) *Football*. (c) *Golden Gate*. (d) *House*. (e) *Peppers2*. (f) *Seven Sisters Cliffs*.

We are particularly interested to see whether X-means could make a better image preprocessing scheme and we compare IAFHA with X-means by performing pre-segmentation experiments. Image *Peppers1* and the six original color images in Section 4.1 are used as test images. We also select several images from the UC-Berkeley Image Segmentation Dataset [48] to carry out

Table 1
MSE test.

Methods	Images	*dc*										
		20	22	24	26	28	30	32	34	36	38	40
IAFHA	*Car*	189.9	**204.6**	222.7	**261.8**	**261.2**	379	**391.1**	390.9	**389.9**	452.3	**452.2**
	Football	179.7	204.9	225.7	215.1	**227.5**	**254.4**	252.5	287.8	**290.9**	450.0	371.4
	Golden_Gate	197.8	221.2	253.0	289.6	285.9	305.2	**326.2**	**385.8**	**438.1**	382.5	**437.7**
	House2	242.3	266.4	266.3	279.9	280.2	279.9	**280.3**	290.7	**290.7**	344.0	343.7
	Peppers2	409.5	436.7	419.1	452.1	**488.6**	**488.5**	488.0	**514.1**	**513.0**	**543.8**	543.2
	S.S.C.	299.5	293.4	326.2	326.9	354.2	**373.8**	358.1	442.5	442.0	**462.6**	502.8
Uniform	*Car*	202.3	267.6	261.6	298.9	320.4	467.7	453.9	403.5	449.7	559.8	487.8
	Football	**169.3**	**189.2**	228.6	203.7	234.4	262.1	252.6	332.3	333.3	527.3	359.9
	Golden_Gate	198.3	206.0	257.0	291.6	298.0	312.4	343.7	385.6	442.1	435.9	420.4
	House2	266.1	253.1	264.3	275.4	269.4	289.0	286.0	307.1	293.0	318.1	**333.8**
	Peppers2	402.9	433.0	414.5	505.9	537.7	530.0	537.5	688.1	541.8	589.4	642.5
	S.S.C.	287.3	293.1	351.3	389.9	370.3	411.3	401.7	444.0	424.9	475.4	592.1
Cluster	*Car*	**187.2**	223.8	**221.6**	265.9	280.5	**340.2**	521.5	**377.3**	503.4	**421.2**	555.7
	Football	174.7	198.6	**222.2**	**203.2**	244.6	258.5	**247.8**	**286.0**	329.7	**446.5**	**369.1**
	Golden_Gate	**180.6**	**192.1**	**236.4**	**270.5**	**265.6**	**303.6**	370.2	399.2	471.7	**382.2**	472.3
	House2	**185.8**	**237.5**	**176.2**	**245.1**	**241.5**	**241.6**	291.3	**241.3**	334.1	**310.5**	345.2
	Peppers2	**369.5**	**389.9**	**404.4**	**444.4**	513.3	548.2	**467.2**	553.6	552.6	564.2	**533.9**
	S.S.C.	**227.6**	**225.3**	**282.6**	**301**	**302.3**	376.6	**327.2**	**394.8**	**399.4**	512.7	**473.4**

Table 2
VPC test.

Methods	Images	*dc*										
		20	22	24	26	28	30	32	34	36	38	40
IAFHA	*Car*	**0.602**	**0.612**	**0.623**	**0.641**	0.640	0.624	**0.698**	**0.697**	**0.698**	**0.749**	**0.749**
	Football	0.584	**0.604**	**0.635**	**0.616**	**0.634**	**0.650**	**0.650**	0.667	**0.666**	0.699	0.672
	Golden_Gate	**0.494**	0.505	0.527	0.547	0.550	**0.562**	**0.569**	**0.615**	**0.618**	**0.611**	**0.621**
	House2	**0.674**	**0.706**	0.707	**0.715**	**0.729**	**0.715**	**0.728**	**0.753**	**0.754**	**0.760**	**0.754**
	Peppers2	**0.478**	**0.499**	**0.497**	**0.513**	**0.525**	**0.526**	**0.526**	**0.541**	**0.539**	**0.547**	**0.547**
	S.S.C.	**0.483**	**0.485**	**0.521**	**0.528**	0.540	**0.550**	**0.551**	0.562	0.561	**0.577**	**0.597**
Uniform	*Car*	0.580	0.579	0.591	0.611	**0.642**	0.585	0.659	0.663	0.640	0.625	0.667
	Football	**0.587**	0.596	0.630	0.611	0.630	0.635	0.648	0.648	0.646	0.678	**0.679**
	Golden_Gate	0.473	0.499	0.519	0.541	0.548	0.553	0.562	0.603	0.608	0.580	0.613
	House2	0.612	0.678	0.652	0.683	0.691	0.619	0.656	0.671	0.740	0.754	0.688
	Peppers2	0.453	0.463	0.476	0.462	0.492	0.492	0.483	0.473	0.494	0.508	0.476
	S.S.C.	0.453	0.451	0.473	0.467	0.488	0.493	0.491	0.552	0.563	0.564	0.548
Cluster	*Car*	0.597	0.599	0.602	0.613	0.618	**0.658**	0.619	0.689	0.662	0.689	0.673
	Football	0.577	0.576	0.622	0.610	0.605	0.634	0.650	**0.667**	0.640	**0.699**	0.674
	Golden_Gate	0.484	**0.509**	**0.528**	**0.557**	**0.555**	0.550	0.559	0.604	0.608	0.602	0.608
	House2	0.652	0.654	**0.729**	0.621	0.664	0.667	0.649	0.710	0.665	0.759	0.712
	Peppers2	0.444	0.495	0.471	0.486	0.496	0.485	0.518	0.499	0.498	0.509	0.524
	S.S.C.	0.463	0.477	0.502	0.504	**0.543**	0.495	0.530	**0.572**	**0.565**	0.539	0.595

extensive tests. Several empirical goodness methods are applied for quantitative evaluation of the segmentation results. These methods estimate the segmentation quality with some human characterization about the properties of "ideal" segmentation and require no prior knowledge of correct segmentation [33]. Here we adopt four evaluation functions to serve as the quantitative benchmarks. The first three functions are respectively:

$F(I)$ proposed by Liu and Yang [41],

$$F(I)=\sqrt{M}\sum_{j=1}^{M}\frac{e_j^2}{\sqrt{N_j}}. \tag{16}$$

$F'(I)$ proposed by Borsotti et al. [42],

$$F'(I)=\frac{1}{1000\cdot N}SQRT\Big(\sum_{a=1}^{MaxArea}[S(a)]^{1+1/a}\Big)\sum_{j=1}^{M}\frac{e_j^2}{\sqrt{N_j}}. \tag{17}$$

$Q(I)$ further refined from $F(I)$ by Borsotti et al. as [42],

$$Q(I)=\frac{1}{1000\cdot N}\sqrt{M}\sum_{j=1}^{M}\left[\frac{e_j^2}{1+\log N_j}+\left(\frac{S(N_j)}{N_j}\right)^2\right]. \tag{18}$$

For the above three formulae, I is the image and N is the total pixel number in I. The segmentation can be described as an assignment of pixels in image I into M regions. Let $\boldsymbol{C}_j$ denote the set of pixels in Region j, and $N_j=|\boldsymbol{C}_j|$ denote the number of pixels in $\boldsymbol{C}_j$. We also define $\boldsymbol{x}_p$ as the feature vector of pixel p and $\boldsymbol{c}_j=(\sum_{p\in\boldsymbol{C}_j}\boldsymbol{c}_p)/N_j$ as the feature centroid of $\boldsymbol{C}_j$. The squared error of cluster j is defined as $e_j^2=\sum_{p\in\boldsymbol{C}_j}(\boldsymbol{x}_p-\boldsymbol{c}_j)^2$. Finally, $S(a)$ denotes the number of regions in image I that have an area of exactly a and *MaxArea* denotes the number of pixels in the largest region.

Notice that all of the above three benchmarks aim at finding a better tradeoff between homogeneity of a region and the total segment number. When segment number increases, the squared

Table 3
VXB test.

Methods	Images	*dc*										
		20	22	24	26	28	30	32	34	36	38	40
IAFHA	*Car*	**0.358**	**0.413**	**0.443**	**0.540**	0.541	0.657	**0.271**	**0.271**	**0.264**	**0.114**	**0.109**
	Football	**0.188**	0.177	**0.135**	**0.183**	**0.122**	**0.120**	0.116	**0.130**	0.137	0.120	0.099
	Golden_Gate	**0.177**	**0.179**	0.229	0.216	0.220	0.248	**0.165**	**0.128**	**0.160**	**0.113**	**0.122**
	House2	**0.366**	**0.190**	0.189	**0.168**	**0.086**	**0.169**	**0.095**	**0.078**	**0.077**	0.101	**0.103**
	Peppers2	**0.529**	**0.412**	**0.458**	**0.452**	**0.543**	**0.541**	**0.578**	**0.381**	**0.425**	**0.432**	**0.424**
	S.S.C.	**0.274**	**0.207**	**0.176**	**0.153**	0.164	**0.200**	**0.180**	0.251	**0.239**	**0.269**	0.258
Uniform	*Car*	0.731	0.885	0.719	0.580	**0.422**	1.300	0.357	0.629	0.883	0.939	0.391
	Football	0.215	**0.172**	0.140	0.212	0.188	0.263	0.119	0.145	**0.126**	0.221	0.167
	Golden_Gate	0.870	0.295	0.248	0.232	0.207	0.243	0.244	0.234	0.227	0.473	0.181
	House2	3.478	0.408	0.848	0.460	0.498	2.584	0.819	0.912	0.095	0.136	2.785
	Peppers2	0.923	0.995	1.007	1.284	0.950	0.839	1.462	1.009	1.334	0.912	1.491
	S.S.C.	0.791	0.734	0.977	1.063	0.841	1.059	0.805	0.321	0.299	0.801	1.211
Cluster	*Car*	0.496	0.516	0.443	0.727	0.653	**0.369**	1.674	0.289	0.428	0.320	0.484
	Football	0.231	0.333	0.198	0.188	0.292	0.234	**0.111**	0.132	0.280	**0.113**	**0.094**
	Golden_Gate	0.429	0.311	**0.158**	**0.178**	**0.181**	**0.203**	0.473	0.181	0.214	0.160	0.192
	House2	3.736	5.117	**0.159**	4.451	4.788	4.777	4.560	0.537	0.825	**0.087**	0.630
	Peppers2	1.948	0.481	1.158	1.131	1.227	1.239	0.590	1.256	1.280	0.717	0.695
	S.S.C.	0.689	0.565	0.719	0.810	**0.163**	1.661	0.854	**0.143**	0.545	1.060	**0.204**

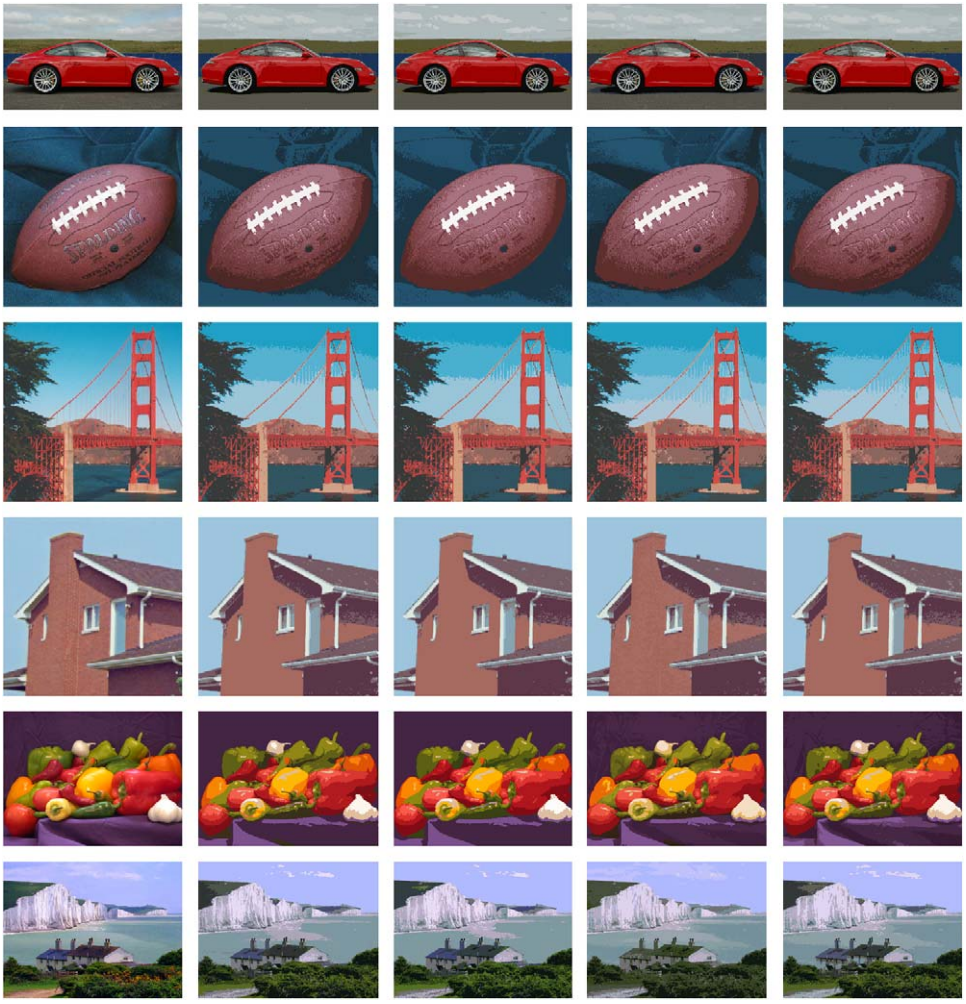

Fig. 8. Pixel clustering results. First column: original images. Second column: AFHA. Third column: FCM. Forth column: IAFHA. Fifth column: AS.

error part, which marks the homogeneity of a region, will generally decrease. An extreme condition is to assign every pixel in an image as a single cluster, reducing squared color error within every cluster to zero. However, this is not likely to happen for usual image segmentation since it obviously makes no sense. The assumption for a good segmentation is to achieve more

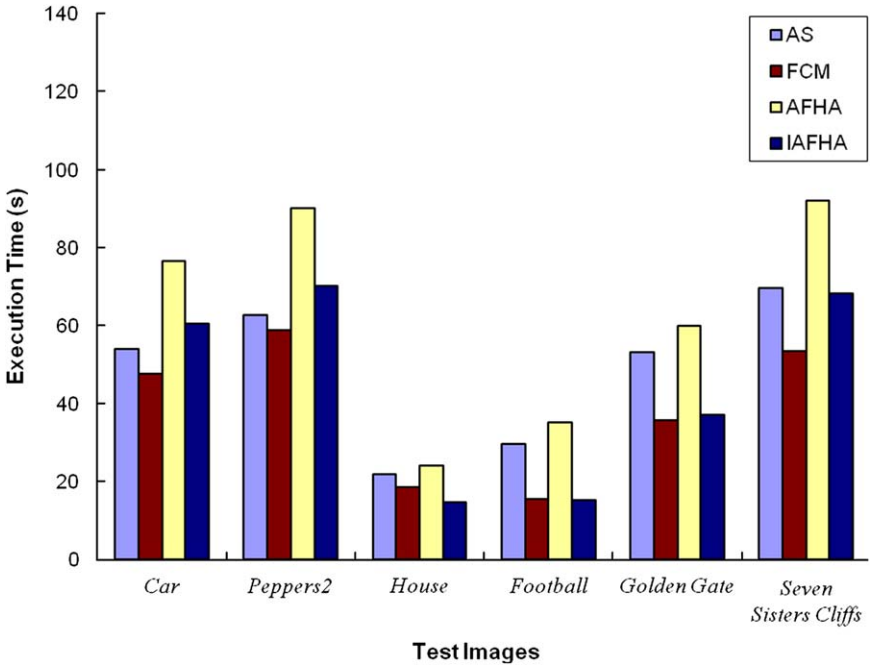

Fig. 9. Illustration of execution time.

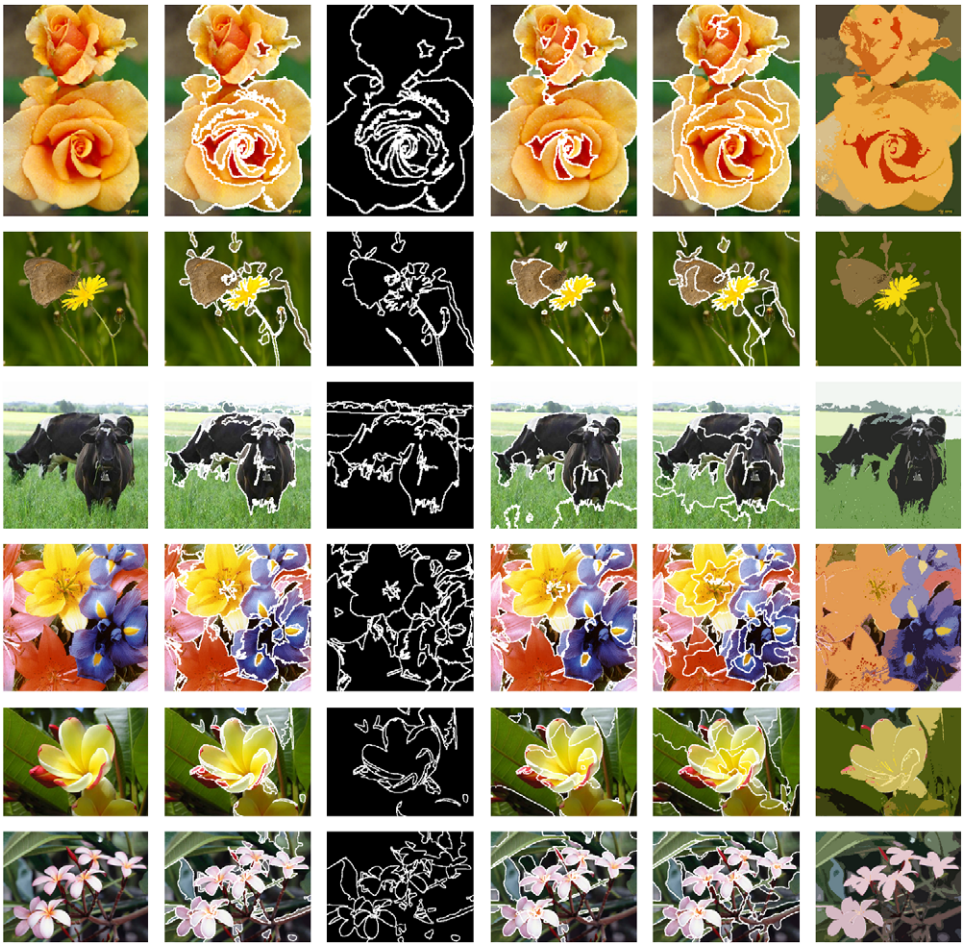

Fig. 10. Image segmentation tests. First column: original images. Second column: IAFHA segmentation. Third column: region boundary of second column. Forth column: MS segmentation. Fifth column: Ncut segmentation. Sixth column: Segmentation using Han and Shi's method. Test images from the first row to the sixth row are *Brandyrose*, *Butterfly*, *Cow*, *Flower*, *Frangipani1*, *Frangipani2*, respectively.

homogeneity within regions while keeping a reasonable segment number.

Apart from these functions, we also adopt a more recently proposed information theoretic approach by Zhang et al. [49]. Their evaluation $E(I)$ is based on entropy and the minimum description length principle (MLD). Given a segmented image, they define V_j as the set of all possible values for the luminance in region j and let $L_j(m)$ denote the number of pixels in region j that have luminance of m in the original image. The entropy for region j is defined as:

$$H(R_j) = -\sum_{m \in Vj} \frac{L_j(m)}{S_j} \log_2 \frac{L_j(m)}{S_j} \tag{19}$$

Then, the expected region entropy of image I is defined as:

$$H_r(I) = -\sum_{j=1}^{N} \left(\frac{S_j}{S_I}\right) H(R_j) \tag{20}$$

Notice that the expected region entropy serves in a similar capacity to the distortion term in F, F′ and Q. Since an over-segmented image will have a very small expected region entropy value, another term which penalizes segmentations should be combined. Thus they also introduced the layout entropy:

$$H_l(I) = -\sum_{j=1}^{N} \left(\frac{S_j}{S_I}\right) \log_2 \frac{S_j}{S_I} \tag{21}$$

The final evaluation $E(I)=H_l(I)+H_r(I)$.

Table 4
Execution time of different algorithms.

Images	Algorithms			
	AS	FCM	AFHA	IAFHA
Car	54.0	57.6	76.6	60.3
Peppers2	62.7	68.7	90.0	70.0
House	22.0	18.7	24.1	14.6
Football	29.7	15.4	35.2	15.3
Golden Gate	53.3	35.8	59.9	37.1
Seven Sisters Cliffs	69.7	53.4	92.1	68.3

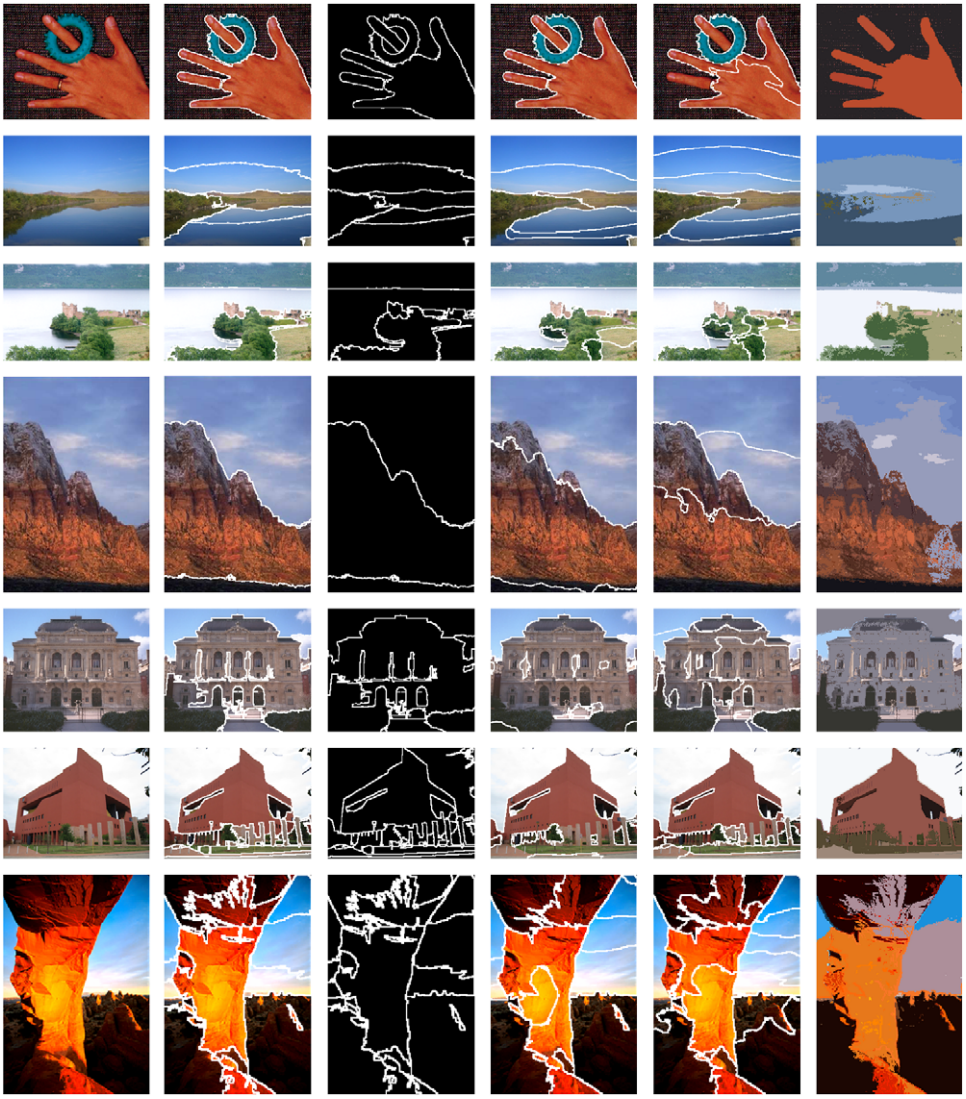

Fig. 11. Image segmentation tests. First column: original images. Second column: IAFHA segmentation. Third column: region boundary of second column. Fourth column: MS segmentation. Fifth column: Ncut segmentation. Sixth column: Segmentation using Han and Shi's method. Test images from the first row to the seventh row are *Hand*, *Lake*, *Loch Ness*, *Mountain*, *Opera*, *Red Building*, *Skyline Arch*, respectively.

Table 5
Quantitative evaluations on pre-segmentations of selected images.

Algorithm and image name		$F(I)$ (*1.0e+8)	$F'(I)$	$Q(I)$ (*1.0e+6)	$E(I)$	Cluster number	Region number
IAFHA	*Car*	0.7329	**17.0077**	**0.0013**	**13.6466**	13	2912
	Football	**0.3367**	**17.8547**	**0.0084**	**14.5556**	10	3626
	Golden Gate	0.6647	**28.2005**	**0.0049**	**15.3514**	16	3847
	House	**0.2168**	**6.3619**	**0.0004**	**13.9399**	10	1816
	Peppers1	**0.6946**	**27.8200**	**0.0030**	**15.3179**	17	3300
	Peppers2	**0.5050**	**10.4700**	**0.0002**	**15.1497**	17	1673
	Seven Sisters Cliff	**0.9647**	**38.2240**	**0.0140**	**14.9257**	19	5544
X-means	*Car*	**0.6419**	46.4751	2.0124	15.2059	123	20861
	Football	0.3789	49.5582	3.9496	15.4142	99	20223
	Golden Gate	**0.4946**	48.9242	0.6393	15.5498	96	12379
	House	0.3736	23.9507	0.0603	14.4628	51	7272
	Peppers1	0.7049	43.9840	0.0460	15.3909	55	6622
	Peppers2	0.5211	23.9640	0.0348	15.6077	129	6836
	Seven Sisters Cliff	1.0094	74.0019	0.6212	15.2874	96	14667

Table 6
Quantitative evaluations on pre-segmentations of selected images from the UC-Berkeley Segmentation Dataset.

Algorithm and image name		$F(I)$ (*1.0e+8)	$F'(I)$	$Q(I)$ (*1.0e+6)	$E(I)$	Cluster number	Region number
IAFHA	*#35008*	**0.3105**	**7.205**	**0.0007**	**14.5313**	14	2403
	#86000	1.7664	**81.0289**	**0.0724**	**15.2212**	20	8772
	#101087	0.7313	**20.9963**	**0.0038**	**11.9827**	13	4089
	#102061	0.8219	**27.1201**	**0.0098**	**14.6467**	16	5272
	#108073	1.3443	**54.6001**	**0.041**	**15.3719**	10	7921
	#176035	0.7200	**26.2581**	**0.0143**	**15.3084**	17	5501
	#208001	1.2351	**46.7883**	**0.0367**	**16.312**	17	8300
	#232038	0.7841	**29.8636**	**0.0244**	**13.6195**	12	6731
	#253036	0.1236	**2.6015**	**0.0003**	**13.3012**	9	1692
	#291000	4.8275	400.8227	**3.6538**	**16.5655**	17	25581
	#295087	1.4928	**61.5692**	**0.0384**	**14.9781**	18	7497
	#302008	0.6508	26.1849	**0.0129**	**11.2466**	13	4529
	#376043	1.9245	**88.9795**	**0.1072**	**16.251**	14	10533
X-means	*#35008*	0.3149	13.0962	0.0307	14.699	60	6639
	#86000	**1.2699**	101.62	1.9225	15.3153	129	19649
	#101087	**0.654**	43.0002	0.6093	12.3943	95	14836
	#102061	**0.4769**	35.182	0.9727	14.8289	105	15794
	#108073	**0.8515**	63.4788	1.3868	15.624	86	18545
	#176035	**0.697**	50.0383	1.1301	15.735	105	17710
	#208001	**1.2283**	50.9847	0.0667	16.3155	30	9747
	#232038	**0.5137**	50.6484	6.6682	14.1725	104	26742
	#253036	**0.099**	4.5366	0.0557	13.9997	115	7725
	#291000	**4.6826**	**390.4325**	3.8903	16.5668	23	26215
	#295087	**1.417**	90.2677	0.6484	14.9988	81	16031
	#302008	**0.2447**	**20.853**	1.4122	11.5498	120	15360
	#376043	**1.5916**	141.7097	6.0314	16.3968	71	29580

Tables 5 and 6 show the evaluations on segmentation of selected images in previous sections and from the UC-Berkeley Segmentation Dataset, respectively. Except $F(I)$, the other three evaluation benchmarks including $F'(I)$, $Q(I)$ and $E(I)$ have favored segmentations by IAFHA. Also notice that $F(I)$ has been reported of being too biased towards over-segmentation with very fine regions [33]. In general, this indicates that IAFHA might be a better approach in term of image preprocessing and segmentation.

Results also show that X-means tends to favor results with larger cluster numbers. We have actually tuned the parameters in IAFHA forcing it to generate the same number of clusters as X-means does, and found that X-means could obtain even better structures. Yet, differences between most results are not very significant and both algorithms have shown their advantage over FCM results that are not properly initialized. This is understandable since IAFHA are not specifically designed for initialization with large number of centroids and the refining of clusters is performed only once with FCM, while in X-means clusters are generated where they are needed and refining is iteratively performed. As a matter of fact, X-means really finds good clustering structures from the pure prospective of machine learning and Bayesian Theory. However, in terms of real world applications, heuristic methods could generate results that are more practical, as indicated by this comparison.

4.6. Applications for color image segmentation

With previous parameter settings and the postprocessing operations described in Section 3.4, we perform color image segmentation using IAFHA. In addition, mean shift, normalized cut (simultaneous K-way cut), Han and Shi's method are implemented for comparison. Region numbers of IAFHA and normalized cut are set to be the same as numbers in mean shift segmentations. It could be indicated from these results that even with the simple dissimilarity measures, IAFHA based segmentation is able to produce relatively satisfactory and accurate segmentations. Moreover, IAFHA tends to preserve more details

Table 7
Comparison on PSNR and postprocessing time.

Image name	PSNR (db)					Processing time (IAFHA)	Processing time (X-means)	Processing time (FCM)	Region number
	Mean shift	IAFHA	X-means	FCM	Ncut				
Brandyrose	17.9807	**18.6093**	17.9303	18.0022	17.9878	**81.3131**	210.8045	96.8366	16
Butterfly	22.6448	24.2724	23.8321	24.0838	**24.3944**	71.3566	395.0590	**68.1658**	16
Cow	21.5597	22.0206	21.9902	21.7599	**22.2428**	272.7347	948.6449	436.7847	23
Flower	17.5670	**18.7185**	17.6634	18.0540	17.1915	248.3126	790.6102	**233.8996**	43
Frangipani1	18.8107	18.9185	18.6730	17.9335	**19.5203**	143.6157	431.6396	**104.4861**	24
Frangipani2	19.6892	18.2396	19.1995	19.3664	**19.4094**	**327.9762**	622.6014	352.3005	49
Hand	21.1485	**21.1809**	21.0576	21.0397	19.5696	**764.9458**	2568.4355	852.1279	5
Lake	22.7476	**24.1123**	20.9394	24.1075	23.4479	20.8511	102.7470	**18.1015**	7
Loch Ness	20.6189	**21.2282**	21.1993	20.9432	20.6019	**51.3326**	482.0608	53.7308	8
Mountain	19.4092	20.6042	20.6032	20.5861	**22.1918**	341.9682	2204.1524	**278.0053**	4
Opera	18.7891	**21.8020**	21.0999	21.5951	21.41389	82.5618	1017.0902	**79.7039**	14
Red Building	22.2231	23.6602	24.0707	22.6129	**24.1084**	62.8192	318.1792	**50.4059**	23
Skyline Arch	**19.3316**	18.5644	14.5872	15.5659	17.6085	86.9771	766.4425	**21.3006**	15
Average	20.1939	**20.9178**	20.2189	20.4346	20.7452	**196.6742**	835.2667	203.5269	19

Table 8
$Q(I)$ evaluation on segmentation results.

Image name	$Q(I)$					
	Mean shift	IAFHA	X-means	FCM	Ncut	Han **& Shi's method**
Brandyrose	1.1102	**1.0816**	1.2402	1.2104	1.3730	11.3464
Butterfly	0.3722	0.2810	0.3087	0.2893	**0.2704**	1.07604
Cow	0.6524	0.6045	**0.5844**	0.6363	0.6001	29.9306
Flower	2.2822	**1.8284**	2.2176	2.0786	2.8978	136.1547
Frangipani1	1.1626	**1.1142**	1.1575	1.4029	1.1634	6.4074
Frangipani2	**1.4674**	2.0538	1.5441	1.5645	1.8278	81.1642
Hand	0.2978	**0.2935**	0.3005	0.3028	0.4372	15.0362
Lake	0.2708	0.2002	0.3901	**0.2001**	0.2377	3.6819
Loch Ness	0.4636	0.4128	**0.4127**	0.4387	0.4991	5.1158
Mountain	0.3810	0.2905	**0.2902**	0.2910	0.3475	98.7471
Opera	0.8570	**0.4656**	0.5218	0.4738	0.5178	118.9640
Red Building	0.5513	0.4460	0.4243	0.5553	**0.4087**	4.5310
Skyline Arch	**0.9241**	1.0937	2.5219	2.1377	1.4542	116.1115
Average	0.8302	**0.7820**	0.9165	0.8909	0.9258	48.3282

along boundaries, which could be potentially utilized for further analysis of textures and refinement of segmentations.

Tables 7 and 8 exhibits the PSNR and $Q(I)$ evaluation results upon segmentations. The same postprocessing operations has been carried out for X-means and randomly initialized FCM. The initial cluster number of FCM is set to be the same as IAFHA. Notice that for both evaluations, segmentations by IAFHA have obtained better results. In addition, the postprocessing speed of IAFHA outperforms X-means, with the postprocessing execution time approximately one third of that of X-means.

5. Conclusions

In this paper, we have introduced an adaptive unsupervised scheme for pixel clustering and color image segmentation. The proposed clustering algorithm called AFHA adaptively initializes cluster centroid distribution and centroid number, showing the advantage of optimization using the Ant System algorithm. Improvements have been made to reduce its computational cost. Test results of clustering structure and execution time demonstrate that ant subsampling could reduce computational complexity while at the same time, exhibiting performances close to those with full number of ants. And our approach could obtain better clustering structures over FCM with random initialization, uniform initialization and preliminary clustering. We have further proposed an IAFHA-based segmentation scheme with spatial information considered. Extensive experiments in which comparison with X-means and segmentation tests are made have demonstrated the practical value of this segmentation method. In conclusion, AFHA could be a feasible preprocessing approach for operations such as image semantic and pattern recognition.

Acknowledgment

The authors would like to express their sincere thanks to the reviewers who made great contributions to the improvement of this paper.

References

[1] G. Economou, A. Fotinos, S. Makrogiannis, S. Fotopoulos, Color image edge detection based on nonparametric estimation, in: Proceedings of the International Conference on Image Processing, 2001, pp. 922–925.
[2] K. Haris, S.N. Efstratiadis, N. Maglaveras, A.K. Katsaggelos, Hybrid image segmentation using watershed and fast region merging, IEEE Trans. Image Process. 7 (12) (1998) 1684–1699.

[3] O. Lezoray, H. Cardot, Cooperation of color pixel classification schemes and color watershed: a study for microscopic images, IEEE Trans. Image Process. 11 (7) (2002) 783–789.
[4] E.J. Pauwels, G. Frederix, Finding salient regions in images, Computer Vision and Image Understanding 75 (1, 2) (1999) 73–85.
[5] Y. Qian, R. Zhao, Image segmentation based on combination of global and local information, in: Proceedings of the International Conference on Image Processing, Santa Barbara, CA., 1997, pp. 204–207.
[6] Y.H. Yang, J. Liu, Multiresolution image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 16 (7) (1994) 689–700.
[7] M. Zhang, L.O. Hall, D.B. Goldgof, A generic knowledge-guided image segmentation and labeling system using fuzzy clustering algorithms, IEEE Trans. Syst. Man Cybern. B Cybern. 32 (5) (2002) 571–582.
[8] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Trans. Pattern Anal. Mach. Intell. 23 (11) (2001) 1222–1239.
[9] P.F. Felzenszwalb, D.P. Huttenlocher, Image segmentation using local variation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA., 1998, pp. 98–103.
[10] Y. Gdalyahu, D. Weinshall, M. Werman, Self-organization in vision: stochastic clustering for image segmentation, perceptual grouping, and image database organization, IEEE Trans. Image Process. 23 (10) (2001) 1053–1074.
[11] D.P. Huttenlocher, G.A. Klanderman, W.J. Rucklidge, Computing images using the Hausdorff distance, IEEE Trans. Pattern Anal. Mach. Intell. 15 (9) (1993) 850–863.
[12] D.W. Jacobs, D. Weinshall, Y. Gdalyahu, Classification with nonmetric distances: image retrieval and class representation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (6) (2000) 583–600.
[13] A.K. Jain, D. Zongker, Representation and recognition of hand-written digits using deformable templates, IEEE Trans. Pattern Anal. Mach. Intell. 19 (12) (1997) 1386–1391.
[14] O.J. Morris, J. Lee, A.G. Constantinides, Graph theory for image analysis: an approach based on the shortest spanning tree, IEE Proc. Part F Commun. Radar Signal Process. 133 (2) (1986) 146–152.
[15] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.
[16] Z. Wu, R. Leahy, An optimal graph theoretic approach to data clustering: theory and its application to image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 15 (11) (1993) 1101–1113.
[17] C.T. Zahn, Graph-theoretical methods for detecting and describing gestalt clusters, IEEE Trans. Comput. C-20 (1) (1971) 68–86.
[18] S. Chiu, Fuzzy model identification based on cluster estimation, J. Intell. Fuzzy Syst. 2 (3) (1994) 267–278.
[19] P. Corsini, B. Lazzerini, F. Marcelloni, A fuzzy relational clustering algorithm based on a dissimilarity measure extracted from data, IEEE Trans. Syst. Man Cybern. B Cybern. 34 (1) (2004) 775–782.
[20] J. Matas, J. Kittler, Spatial and feature space clustering: application in image analysis, in: Proceedings of the 6th International Conference on Computer Analysis Images Patterns, 1995, pp. 162–173.
[21] E. Pekalska, R.P.W. Duin, Dissimilarity representations allow for building good classifiers, Pattern Recognition Lett. 23 (8) (2002) 943–956.
[22] S.J. Roberts, Parametric and nonparametric unsupervised cluster analysis, Pattern Recognition 30 (2) (1997) 261–272.
[23] R. Schumeyer, K. Barner, A color-based classifier for region identification in video, in: Proceedings of Visual Communications and Image Processing 3309 (1998) 189–200.
[24] R. Yager, D. Filev, Generation of fuzzy rules by mountain clustering, J. Intell. Fuzzy Syst. 2 (3) (1994) 209–219.
[25] J. Xu, P.F. Shi, Natural color image segmentation using integrated mechanism, Chin. Opt. Lett. 1 (11) (2003) 45–47.
[26] A.R. Malisia, H.R. Tizhoosh, Image thresholding using Ant Colony Optimization, in: Proceedings of the 3rd Canadian Conference on Computer and Robot Vision, 2006, p. 26.
[27] N. Yumusak, F. Temurtas, O. Cerezci, S. Pazar, Image thresholding using measures of fuzziness, in: Proceedings of the 24th Annual Conference of the IEEE, 1998, pp. 1300–1305.
[28] Y.F. Han, P.F. Shi, An improved ant colony algorithm for fuzzy clustering in image segmentation, Neurocomputing 70 (2007) 665–671.
[29] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619.
[30] W.B. Tao, H. Jin, Y.M. Zhang, Color image segmentation based on mean shift and normalized cuts, IEEE Trans. Syst. Man Cybern. B Cybern. 37 (5) (2007) 1382–1389.
[31] S. Makrogiannis, G. Economou, S. Fotopoulos, A region dissimilarity relation that combines feature-space and spatial information for color image segmentation, IEEE Trans. Syst. Man Cybern. B Cybern. 35 (1) (2005) 44–53.
[32] V. Grau, A.U.J. Mewes, et al., Improved watershed transform for medical image segmentation using prior information, IEEE Trans. Med. Imag. 23 (4) (2004) 447–458.
[33] H. Zhang, J.E. Fritts, S.A. Goldman, A co-evaluation framework for improving segmentation, in: Proceedings of the SPIE, vol. 5809, 2005, pp. 420–430.
[34] E. Ruspini, Numerical methods for fuzzy clustering, Inf. Sci. 2 (1970) 319–350.
[35] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact, well separated clusters, Cybernetics 3 (1974) 95–104.
[36] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.
[37] S.S. Reddi, S.F. Rudin, H.R. Keshavan, An optimal multiple threshold scheme for image segmentation, IEEE–SMC 14 (1984) 611–665.
[38] A. Colorni, M. Dorigo, V. Maniezzo, et al., Distributed optimization by ant colonies, in: Proceedings of the 1st European Conference on Artificial Life, 1991, pp. 134–142.
[39] M. Dorigo, T. Stutzle, Ant Colony Optimization, The MIT Press, Cambridge, MA, USA, pp. 70–75.
[41] J. Liu, Y.H. Yang, Multi-resolution color image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 16 (1994) 689–700.
[42] M. Borsotti, P. Campadelli, R. Schettini, Quantitative evaluation of color image segmentation results, Pattern Recognition Lett. 19 (1998) 741–747.
[43] L.G. Shapiro, G.C. Stockman, Computer Vision, Prentice-Hall, New Jersey, 2001 pp. 279–325.
[44] X. Ren, J. Malik, Learning a Classification Model for Segmentation, in: Proceedings of the 9th International Conference on Computer Vision, 2003, pp. 10–17.
[45] J.C. Bezdek, Cluster validity with fuzzy sets, Cybern. Syst. 3 (1974) 58–73.
[46] J.C. Bezdek, Mathematical models for systematics and taxonomy, in: Proceedings of 8th International Conference on Numerical Taxonomy, San Francisco, CA, 1975, pp. 143–166.
[47] X.L. Xie, G.A. Beni, Validity measure for fuzzy clustering, IEEE Trans. Pattern Anal. Mach. Intell. 3 (1991) 841–846.
[48] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Proceedings of 8th International Conference on Computer Vision, 2001, pp. 416–423.
[49] H. Zhang, J. Fritts, S. Goldman, An entropy-based objective evaluation method for image segmentation, in: Proceedings of SPIE- Storage and Retrieval Methods and Applications for Multimedia, 2004, pp. 38–49.
[50] M.N. Ahmed, S.M. Yamany, N. Mohamed, A.A. Farag, A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data, IEEE Trans. Med. Imag. 3 (2002) 193–199.
[51] S. Khan, A. Ahmad, Cluster center initialization algorithm for K-means clustering, Pattern Recognition Lett. 25 (2004) 1293–1302.
[52] D. Pelleg, A. Moore, X-means: extending k-means with efficient estimation of the number of clusters, in: Proceedings of 7th International Conference on Machine Learning, 2000, pp. 727–734.
[53] Y.F. Han, P.F. Shi, An ant colony algorithm based image segmentation method, Comput. Eng. Appl. (2004) 5–7.

About the Author—ZHIDING YU received the B.Eng. degree in Information Engineering from South China University of Technology and the M.Sc. degree in Electronic Engineering from The Hong Kong University of Science and Technology. He is now an M.Phil. candidate at Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology. His current research interests include multimedia signal processing, machine learning and computer vision.

About the Author—OSCAR C. Au received his B.A.Sc. from University of Toronto in 1986, his M.A. and Ph.D. from Princeton University in 1988 and 1991, respectively. After being a postdoctoral researcher in Princeton University for one year, he joined the Hong Kong University of Science and Technology (HKUST) as an Assistant Professor in 1992. He is/has been an Associate Professor of the Department of Electronic and Computer Engineering, Director of Multimedia Technology Research Center (MTrec), and Director of the Computer Engineering (CPEG) Program in HKUST.

His main research contributions are on video and image coding and processing, watermarking and light weight encryption, speech and audio processing. Research topics include fast motion estimation for MPEG-1/2/4, H.261/3/4 and AVS, optimal and fast sub-optimal rate control, mode decision, transcoding, denoising, deinterlacing, post-processing, multi-view coding, scalable video coding, distributed video coding, subpixel rendering, JPEG/JPEG2000, HDR imaging, compressive sensing, halftone image data hiding, GPU-processing, software-hardware co-design, etc. He has published about 260 technical journals and conference papers. His fast motion estimation algorithms were accepted into the ISO/IEC 14496-7 MPEG-4 international video coding standard and the China AVS-M standard. His light-weight encryption and error resilience algorithms are accepted into the AVS standard. He has 3 US patents and is applying for 50+ more on his signal processing techniques. He has performed forensic investigation and stood as an expert witness in the Hong Kong courts many times.

About the Author—RUOBING ZOU is currently pursuing her B.Eng. degree at the School of Electronic and Information Engineering, South China University of Technology.

About the Author—WEIYU YU received the B.Eng. degree in Electronical Automation from Guangdong University of Technology, the M.Eng. degree in Electronic and Information Engineering from South China University of Technology and the Ph.D. degree (Electrical and Electronic Engineering), School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China.

His research interests include image processing, video analysis and pattern recognition.

About the Author—JING TIAN received the B.Eng., M.Eng. degrees, all in Electronic and Information Engineering, from South China University of Technology, and the Ph.D. degree in Electrical and Electronic Engineering from Nanyang Technological University. His research interests include image processing, pattern recognition and computer vision.