Pattern Recognition 44 (2011) 777–787

Contents lists available at ScienceDirect

Pattern Recognition

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

Color image segmentation using pixel wise support vector machine classification

Xiang-Yang Wang [a,b,*], Ting Wang [a], Juan Bu [a]

[a] School of Computer and Information Technology, Liaoning Normal University, Dalian 116029, China
[b] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

ARTICLE INFO

Article history:
Received 11 March 2010
Received in revised form
9 June 2010
Accepted 7 August 2010

Keywords:
Image segmentation
Support vector machine
Fuzzy c-means
Local homogeneity model
Gabor filter

ABSTRACT

Image segmentation is an important tool in image processing and can serve as an efficient front end to sophisticated algorithms and thereby simplify subsequent processing. In this paper, we present a color image segmentation using pixel wise support vector machine (SVM) classification. Firstly, the pixel-level color feature and texture feature of the image, which is used as input of SVM model (classifier), are extracted via the local homogeneity model and Gabor filter. Then, the SVM model (classifier) is trained by using FCM with the extracted pixel-level features. Finally, the color image is segmented with the trained SVM model (classifier). This image segmentation not only can fully take advantage of the local information of color image, but also the ability of SVM classifier. Experimental evidence shows that the proposed method has a very effective segmentation results and computational behavior, and decreases the time and increases the quality of color image segmentation in comparison with the state-of-the-art segmentation methods recently proposed in the literature.

1. Introduction

Image segmentation is a classic inverse problem, which consists of achieving a compact region-based description of the image scene by decomposing it into meaningful or spatially coherent regions sharing similar attributes. This low-level vision task is often the preliminary (and also crucial) step in many video and computer vision applications, such as object localization or recognition, data compression, tracking, image retrieval, or understanding. In recent years, a number of very inspiring and pioneering image segmentation algorithms have been developed, and these algorithms can be roughly classified into five major categories [1,2]: thresholding [3,4], template matching [5,6], clustering [7–9], edge detection [10–12] and region growing [13–15]. These algorithms have been proven to be successful in many applications, but none of them are generally applicable to all images and different algorithms are usually not equally suitable for a particular application.

Image thresholding methods are popular due to their simplicity and efficiency. However, traditional histogram-based thresholding algorithms cannot separate those areas which have the same gray level but do not belong to the same part. In addition, they cannot process images whose histograms are nearly unimodal, especially when the target region is much smaller than the background area. Template matching method becomes time consuming when the image becomes more complex or larger in size. Clustering method, viewing an image as a set of multi-dimensional data and classifying the image into different parts according to certain homogeneity criterion, can get much better results of segmentation. But over-segmentation is the problem that must be settled and feature extraction is an important factor for clustering. The edge detection method is one of the widely used approaches to the problem of image segmentation. It is based on the detection of points with abrupt changes at gray levels. The main disadvantages of the edge detection technique are that it does not work well when images have many edges, and it cannot easily identify a closed curve or boundary. Region growing algorithms deal with spatial repartition of the image feature information. In general, they perform better than the thresholding approaches for several sets of images. However, the typical region growing processes are inherently sequential. The regions produced depend both on the order in which pixels are scanned and on the value of pixels which are first scanned and gathered to define each new segment. In view of the problems mentioned above, plenty of approaches and their corresponding improvements have been proposed to ensure the accuracy and rapidity of image segmentation. But there is still much work to be done to overcome their drawbacks, and attempts at utilizing knowledge on other domains, especially artificial intelligence, should be highly appreciated [2].

Recently, intelligent approaches, such as neural network and support vector machine (SVM) [16], have already been utilized

* Corresponding author at: School of Computer and Information Technology, Liaoning Normal University, Dalian 116029, China. Tel.: +86 411 85992415; fax: +86 411 85992323.
E-mail address: wxy37@126.com (X.-Y. Wang).

0031-3203/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2010.08.008

successfully in image segmentation. Quan and Wen [17] proposed an effective multiscale method for the segmentation of the synthetic aperture radar (SAR) images via probabilistic neural network. By combining the probabilistic neural network (PNN) with the multiscale autoregressive (MAR) model, a classifier, which inherits the excellent strongpoint from both of them, is designed. Yu and Chang [18] presented an effective and efficient method for solving scenery image segmentation by applying the SVMs methodology. In scheme [19], the problem of scarcity of labeled pixels, required for segmentation of remotely sensed satellite images in supervised pixel classification framework, is addressed. Yan and Zheng [20] proposed a SAR image segmentation method based on one-class SVM, in which one-class SVM and two-class SVM for segmentation are discussed. Cyganek [21] proposed an efficient color segmentation method which is based on the SVM classifier operating in a one-class mode, and the method has been developed especially for the road signs recognition system. In scheme [22], support vector clustering (SVC) is used for marketing segmentation, and a case study of a drink company is used to demonstrate the proposed method and compared with the *k*-means and the self-organizing feature map (SOFM) methods. Ji et al. [23] proposed a new semi-supervised approach based on transductive support vector machine (TSVM) to segment SAR images, and it is robust to noises and is effective when dealing with low numbers of high-dimensional samples. Xue et al. [24] presented a method of automatic segmentation and classification of mosaic patterns in cervigrams in which a support vector machine (SVM) classifier is applied, using learning from a "ground truth" dataset annotated by medical experts in oncology and gynecology. In the approach, the acetowhite region is split into tiles, and texture features are extracted from each tile. The SVM classifier is trained using the texture features of tiles obtained from ground truth images. Given a new test image, the trained SVM classifier is applied to classify each tile in the test image, and the classified tiles are combined to generate the final segmentation map.

In this paper, we propose a color image segmentation using pixel wise support vector machine (SVM) classification. Firstly, the pixel-level color feature and texture feature of the image, which is used as input of SVM model (classifier), are extracted via the local homogeneity model and Gabor filter. Then, the SVM model (classifier) is trained by using FCM with the extracted pixel-level features. Finally, the color image is segmented with the trained SVM model (classifier). This image segmentation not only can fully take advantage of the local information of color image, but also the ability of SVM classifier. Simulation results show that the proposed method achieves competitive segmentation results compared to the state-of-the-art segmentation methods recently proposed in the literature.

The rest of this paper is organized as follows. Section 2 presents the basic theory about SVM. In Section 3, the pixel-level color feature and texture feature extraction are described. Section 4 contains the description of our color image segmentation. Simulation results in Section 5 will show the performance of our scheme. Finally, Section 6 concludes this presentation.

2. The support vector machine (SVM)

Support vector machines (SVM) have been successfully applied in classification and function estimation problems after their introduction by Vapnik within the context of statistical learning theory and structural risk minimization [25]. Vapnik constructed the standard SVM to separate training data into two classes. The goal of the SVM is to find the hyper-plane that maximizes the minimum distance between any data point, as shown in Fig. 1.

Given a training dataset of l points $\{x_i,y_i\}_{i=1}^{l}$ with the input data $x_i \in \Re^n$ and the corresponding target $y_i \in \{-1,+1\}$. In feature space SVM models take the form

$$\mathbf{y}(\mathbf{x}) = \boldsymbol{\omega}^{\mathbf{T}} \varphi(\mathbf{x}) + \mathbf{b} \tag{1}$$

where the nonlinear mapping $\phi(\cdot)$ maps the input vector into a so-called higher dimensional feature space, $\mathbf{b}$ is the bias and $\boldsymbol{\omega}$ is a weight vector of the same dimension as the feature space.

SVM formulations start from the assumption that the linear separable case is

$$\begin{cases} \boldsymbol{\omega}^T x_i + \mathbf{b} \geq +1 & \text{if } y_i = +1 \\ \boldsymbol{\omega}^T x_i + \mathbf{b} \leq -1 & \text{if } y_i = -1 \end{cases} \tag{2}$$

For the non-separable case

$$\begin{cases} \boldsymbol{\omega}^T \boldsymbol{\varphi}(x_i) + \mathbf{b} \geq +1 & \text{if } y_i = +1 \\ \boldsymbol{\omega}^T \boldsymbol{\varphi}(x_i) + \mathbf{b} \leq -1 & \text{if } y_i = -1 \end{cases} \tag{3}$$

In this space, a linear decision surface is constructed with special properties that ensure high generalization ability of the network. By use of a nonlinear kernel function, it is possible to compute a separating hyper-plane with a maximum margin in a feature space.

We need to find, among all hyper-planes separating the data, an existing maximum margin $2/||\boldsymbol{\omega}||$ between the classes.

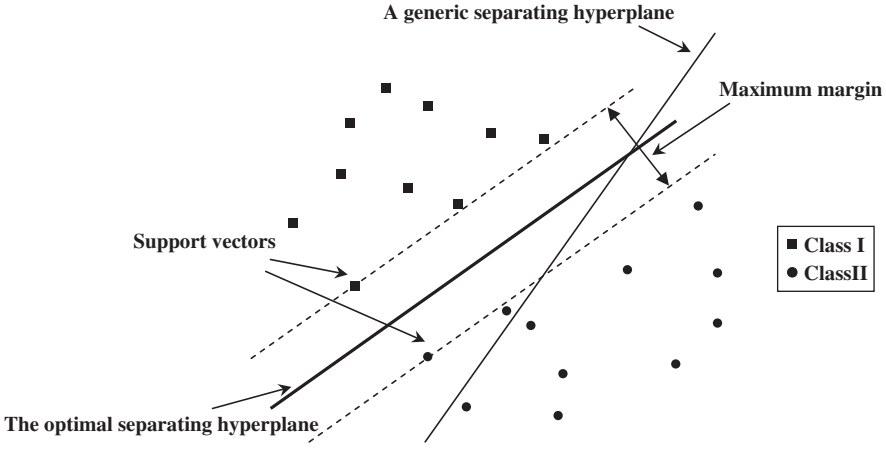

Fig. 1. The basic theory of SVM (linear separation).

The problem is transformed into a quadratic programming problem

$$\min \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega}+C\sum_{i=1}^{l}\zeta_i$$
$$\text{s.t } y_i(\boldsymbol{\omega}^T\boldsymbol{\varphi}(x_i)+\mathbf{b})=1-\zeta_i,\quad \zeta_i\geq 0, i=1,\ldots,l \tag{4}$$

where C is the trade-off parameter between the error and margin.

The quadratic programming problem can be solved by using Lagrangian multipliers $\alpha_i\in\Re$. The solution satisfies the Karush–Kuhn–Tucker (KKT) conditions. $\boldsymbol{\omega}$ can be recovered by using $\boldsymbol{\omega}=\sum_{i=1}^{l}\alpha_i y_i\boldsymbol{\varphi}(x_i)$, where x_i are non-zero values and α_i are support vectors (SV).

The decision boundary is determined only by the SV. Let $t_j\,(j=1,\ldots,s)$ be the indices of the s support vectors. Then we can rewrite

$$\boldsymbol{\omega}=\sum_{j=1}^{s}\alpha_{t_j}y_{t_j}\boldsymbol{\varphi}(x_{t_j}) \tag{5}$$

The quadratic programming problem is solved by considering the dual problem

$$\max_{\alpha} Q(\alpha)=-\frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j K(x_i,x_j)$$
$$\text{s.t}\quad \begin{cases} 0\leq\alpha_i\leq C \quad \forall i \\ \sum_{i=1}^{l}\alpha_i y_i=0 \end{cases} \tag{6}$$

With the kernel trick (Mercer Theorem)

$$K(x_i,x_j)=\varphi(x_i)^T\varphi(x_j) \tag{7}$$

Several types of kernels, such as linear, polynomial, splines, RBF, and MLP, can be used within the SVM. This finally results in the following:

$$\mathbf{y}(x)=\text{sign}\left(\sum\alpha_i y_i K(x,x_i)+b\right) \tag{8}$$

In addition to linear classification, SVM can be applied to nonlinear classification problems. When applying SVM in nonlinear problems, nonlinear mapping is used to generate the classification features from the original data. The nonlinearly separable data to be classified is mapped onto a high-dimensional feature space, where the data can be linearly classified [26].

3. The pixel-level color and texture feature

In this paper, each pixel of an image is identified as belonging to a homogenous region corresponding to an object or part of an object. The problem of image segmentation is regarded as a classification task, and the goal of segmentation is to assign a label to individual pixel or a region. So, it is very important to extract the effective pixel-level image feature. Here, we extract the pixel-level color and texture feature via the local homogeneity model and Gabor filter.

3.1. Pixel color feature

Color is one of the most dominant and distinguishable low-level visual features in describing image, and many researchers have employed color to segment images [2]. In this section, we introduce a new pixel-level color feature according to local homogeneity model.

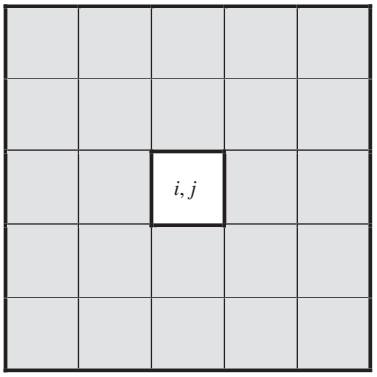

Fig. 2. Local image window.

Here, we will select the CIE L*a*b* (CIELAB) color space to extract pixel-level color feature, because CIE L*a*b* space has metric color difference sensitivity to a good approximation and is very convenient to measure small color difference, while the RGB color space does not [2]. Let $P_{ij}=(P_{ij}^L,P_{ij}^a,P_{ij}^b)$ represent the three color components of a pixel at the location (i,j) in an M × N image, and the pixel-level color feature $CF_{ij}^k(k=L,a,b)$ of color component $P_{ij}^k(k=L,a,b)$ can be computed as follows.

(1) *Construct the local image window*
As shown in Fig. 2, w_{ij} is a size $d\times d$ (for example, 5×5) window centered at (i,j) for the computation of pixel-level color feature, and d is odd integer greater than 1.

(2) *Compute the pixel color feature of component* P_{ij}^k
Homogeneity is largely related to the local information extracted from an image and reflects how uniform a region is [1]. It plays an important role in image segmentation since the result of image segmentation would be several homogeneous regions. We define the local homogeneity as the pixel-level color feature, which consists of two components: standard deviation and discontinuity of the color component P_{ij}^k.

The standard deviation of color component $P_{ij}^k(k=L,a,b)$ is calculated as

$$v_{ij}^k=\sqrt{\frac{1}{d^2}\sum_{m=i-((d-1)/2)}^{i+((d-1)/2)}\sum_{n=j-((d-1)/2)}^{j+((d-1)/2)}(P_{mn}^k-\mu_{ij}^k)^2} \tag{9}$$

where $0\leq i,m\leq M-1, 0\leq j,n\leq N-1$.

μ_{ij}^k is the mean of the color component $P_{ij}^k(k=L,a,b)$ within window w_{ij}, and calculated as

$$\mu_{ij}^k=\frac{1}{d^2}\sum_{m=i-((d-1)/2)}^{i+((d-1)/2)}\sum_{n=j-((d-1)/2)}^{j+((d-1)/2)}P_{mn}^k \tag{10}$$

The discontinuity for color component P_{ij}^k is described by edge value. There are many different edge operators: Sobel, Laplace, Canny, etc. Since we do not need to find the exact locations of the edges, and due to its simplicity, we employ Sobel operator to calculate the discontinuity and use the magnitude $e_{ij}^k(k=L,a,b)$ of the gradient at location (i,j) as the measurement:

$$e_{ij}^k=\sqrt{G_x^{k2}+G_y^{k2}} \tag{11}$$

where G_x^k and G_y^k are the components of the gradient of color component $P_{ij}^k(k=L,a,b)$ in the x and y directions, respectively.

The standard deviation and discontinuity values are normalized in order to achieve computational consistence:

$$V_{ij}^k = \frac{v_{ij}^k}{v_{max}^k}, \quad E_{ij}^k = \frac{e_{ij}^k}{e_{max}^k} \tag{12}$$

where $v_{max}^k = \max\{v_{ij}^k\}, e_{max}^k = \max\{e_{ij}^k\}$, $(0 \le i \le M-1, 0 \le j \le N-1)$, $(k = L,a,b)$

The local homogeneity is represented as

$$CF_{ij}^k = H_{ij}^k = 1 - E_{ij}^k \times V_{ij}^k \tag{13}$$

where $0 \le i \le M-1, 0 \le j \le N-1$, $k = L,a,b$.

The value of the local homogeneity at each location of an image has a range from 0 to 1. The more uniform the local region surrounding a pixel is, the larger the local homogeneity value the pixel has. The size of the windows has influence on the calculation of the local homogeneity value. The window should be big enough to allow enough local information to be involved in the computation of the local homogeneity for the pixel. Furthermore, using a larger window in the computation of the local homogeneity increases smoothing effect, and makes the derivative operations less sensitive to noise. However, smoothing the local area might hide some abrupt changes of the local region. Also, a large window causes significant processing time. Weighing the pros and cons, we choose a 5×5 window for computing the standard deviation and discontinuity.

So, we can obtain the pixel-level color feature CF_{ij} of the image pixel P_{ij} at location (i,j)

$$CF_{ij} = (H_{ij}^L, H_{ij}^a, H_{ij}^b) \tag{14}$$

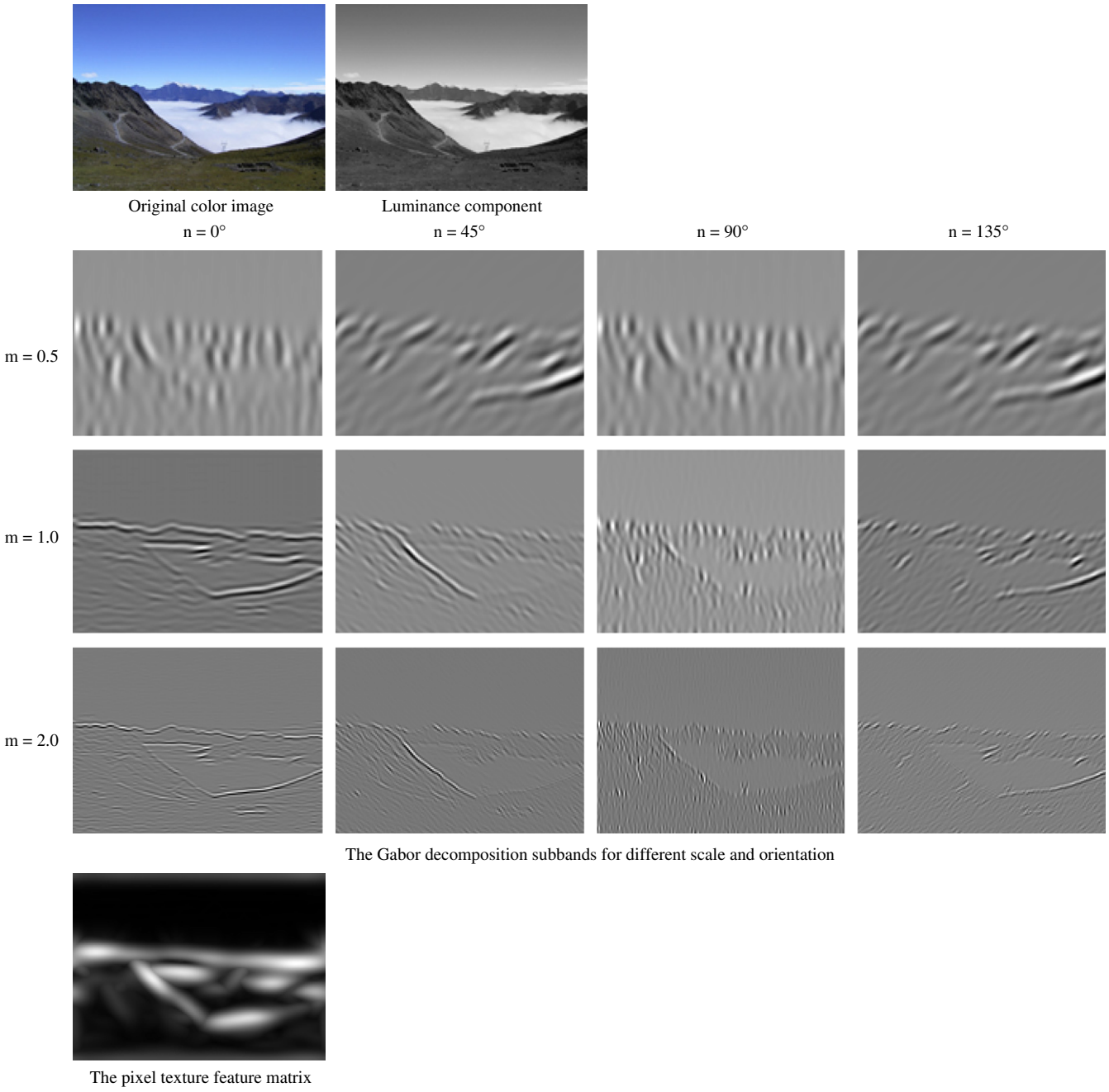

Fig. 3. The pixel texture feature extraction.

3.2. Pixel texture feature

Texture is one common feature used in image segmentation. It is often used in conjunction with color information to achieve better segmentation results than possible with just color alone. To obtain the pixel-level texture feature, we apply the Gabor filter to the image, and extract the local energy of the filter responses, which is regarded as the pixel texture feature.

Gabor filter [27] is a class of filters in which a filter of arbitrary orientation and scale is synthesized as a linear combination of a set of "basis filters". The edge located at different orientations and scales in an image can be detected by splitting the image into orientation and scale subbands obtained by the basis filters having these orientations and scales. It allows one to adaptively "steer" a filter to any orientation and scale, and to determine analytically the filter output as a function of orientation and scale.

A two dimensional Gabor filter $g(x,y)$ is an oriented sinusoidal grating, which is modulated by a two dimensional Gaussian function $h(x,y)$ as follows:

$$g(x,y) = h(x,y)\exp(2\pi jWx) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x{}^2}+\frac{y^2}{\sigma_y{}^2}\right)\right]\exp(2\pi jWx) \quad (15)$$

and the Fourier transform $G(u,v)$ of $g(x,y)$ is given by

$$G(u,v) = H(u-W,v) = \exp\left\{-\frac{1}{2}\left[\frac{(u-W)^2}{\sigma_u^2}+\frac{v^2}{\sigma_v^2}\right]\right\} \quad (16)$$

Here $h(x,y)$ is a two dimensional Gaussian function with its center at the origin, and σ_x and σ_y denote its variances in x and y directions, respectively. The variances of the Fourier transform function $G(u,v)$ are $\sigma_u = 1/2\pi\sigma_x$ and $\sigma_v = 1/2\pi\sigma_y$. And W is the modulation frequency.

For the mother Gabor filter $g(x,y)$, its children Gabor filters $g_{mn}(x,y)$ are defined to be its scaled and rotated versions:

$$g_{mn}(x,y) = a^{-2m}g(x',y'), a \geq 1$$
$$\begin{pmatrix} x' \\ y' \end{pmatrix} = a^{-m}\begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}$$
$$\theta = \frac{n\pi}{L};\ m = 0,1,\ldots,K-1;\ n = 0,1,\ldots,L-1 \quad (17)$$

where a is a fixed scale factor, m is the scale parameter, n is the orientation parameter, K is the total number of scales, and L is the total number of orientations. In this paper, we set the Gabor function parameters as follows: $W=1$, $a=2$, $\sigma_x=\sigma_y=1/2\pi$, $K=3$, $L=4$.

Let $I(x,y)$ denote an image with size $w \times h$. The Gabor-filtered output $G_{mn}(x,y)$ of the image $I(x,y)$ is defined as its convolution with the Gabor filter $g_{mn}(x,y)$:

$$G_{mn}(x,y) = I(x,y) * g_{mn}(x,y) = \sum_{x_1=1}^{w}\sum_{y_1=1}^{h} I(x-x_1,y-y_1)g_{mn}(x_1,y_1) \quad (18)$$

Here, $G_{mn}(x,y)$ is the Gabor-filtered image at the scale parameter m and the orientation parameter n.

The Fourier transform function $\hat{G}_{mn}(u,v)$ of the Gabor filter output $G_{mn}(x,y)$ is given by

$$\hat{G}_{mn}(u,v) = \hat{I}(u,v) \times \hat{g}_{mn}(u,v) \quad (19)$$

where $\hat{I}(u,v)$ is the Fourier transform function of the image $I(x,y)$, and $\hat{G}_{mn}(u,v)$ is the Fourier transform function of the Gabor filter $G_{mn}(x,y)$.

In this paper, we base our pixel texture feature extraction on Gabor filter, which can be designed to produce any number of orientation and scale bands. One of the most commonly used features for texture analysis is the energy of the subband coefficients. Various nonlinear operations have been used to boost up the sparse subband coefficients. Our approach is based on the local median energy of the subband coefficients. The advantage of the median filter is that it suppresses textures associated with transitions between regions, while it responds to texture within uniform regions.

The main steps of pixel-level texture feature extracting procedure developed can be described as follows:

Step 1. *Color space transformation*

The color image I is transformed from RGB color space to YCbCr color space. Here, Y is the luminance component, and Cb and Cr are the chrominance components.

Step 2: *Applying the Gabor filter to the luminance component Y*

We use Gabor filter decomposition with 4 orientation and 3 scale subbands. Most researchers have used 4–6 orientation bands to approximate the orientation selectivity of the human visual system. Since the images are fairly small, we found that a 3 level decomposition is adequate. Out of those we use only the 4 orientation and 3 scale bands. Our goal is to identify regions with the dominant orientation (0°, 45°, 90°, 135°), and it is the maximum of the 12 coefficients that determines the orientation at a given pixel location. As shown in Fig. 3.

Step 3. *Pixel texture feature extraction*

Let $B_{ij}^{m,n}$ represent the Gabor subband coefficient at location (i,j) that corresponds to the $m(m=0.5, 1.0, 2.0)$ scale and $n(n=0°, 45°, 90°, 135°)$ orientation. We will use TF_{ij} to denote the maximum (in absolute value) of the 12 coefficients at location (i,j), which is the pixel texture feature at location (i,j).

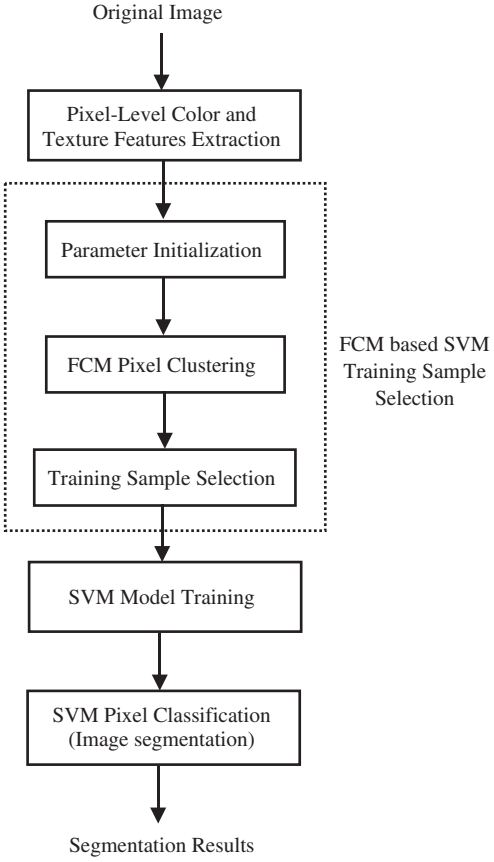

Fig. 4. The pixel-based color image segmentation using SVM and FCM.

4. The pixel-based color image segmentation using SVM and FCM

We know that the image segmentation can be taken as classification problems, which can be solved using anyone of well-known classification techniques. SVM is one of the classification techniques and good results of the SVM technique in pattern recognition have been obtained, so we can choose the SVM for solving color image segmentation problems. In this paper, we present a pixel-based color image segmentation using SVM and FCM. Firstly, the pixel-level color feature and texture feature of the image, which are used as input of SVM model

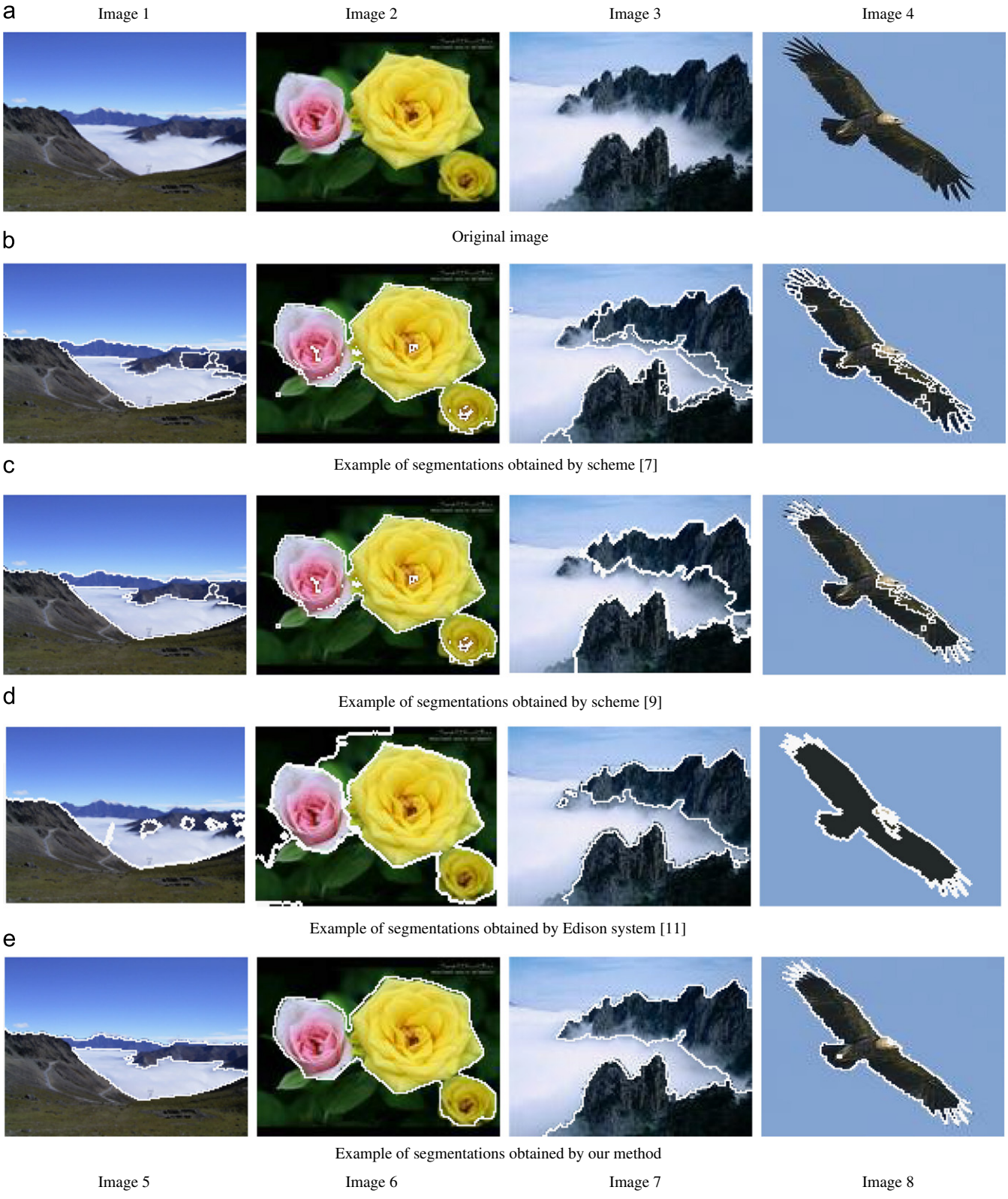

Fig. 5. Segmentation example obtained by four different methods (images 1–4).

(classifier), are extracted via the local homogeneity model and Gabor filter. Then, the SVM model (classifier) is trained by using FCM with the extracted pixel-level features. Finally, the color image is segmented with the trained SVM model (classifier). This image segmentation not only can fully take advantage of the local information of color image, but also the ability of SVM classifier.

The pixel-based color image segmentation using SVM and FCM can be summarized as follows (see Fig. 4):

(1) *Pixel-level color and texture features extraction*
The pixel-level color and texture features are extracted via the local homogeneity model and Gabor filter (see Section 3).

(2) *FCM based SVM training sample selection*
Training sample selection is one of the major factors determining to what degree the SVM classification rules can be generalized to unseen sample. A previous study showed that this factor could be more important for obtaining

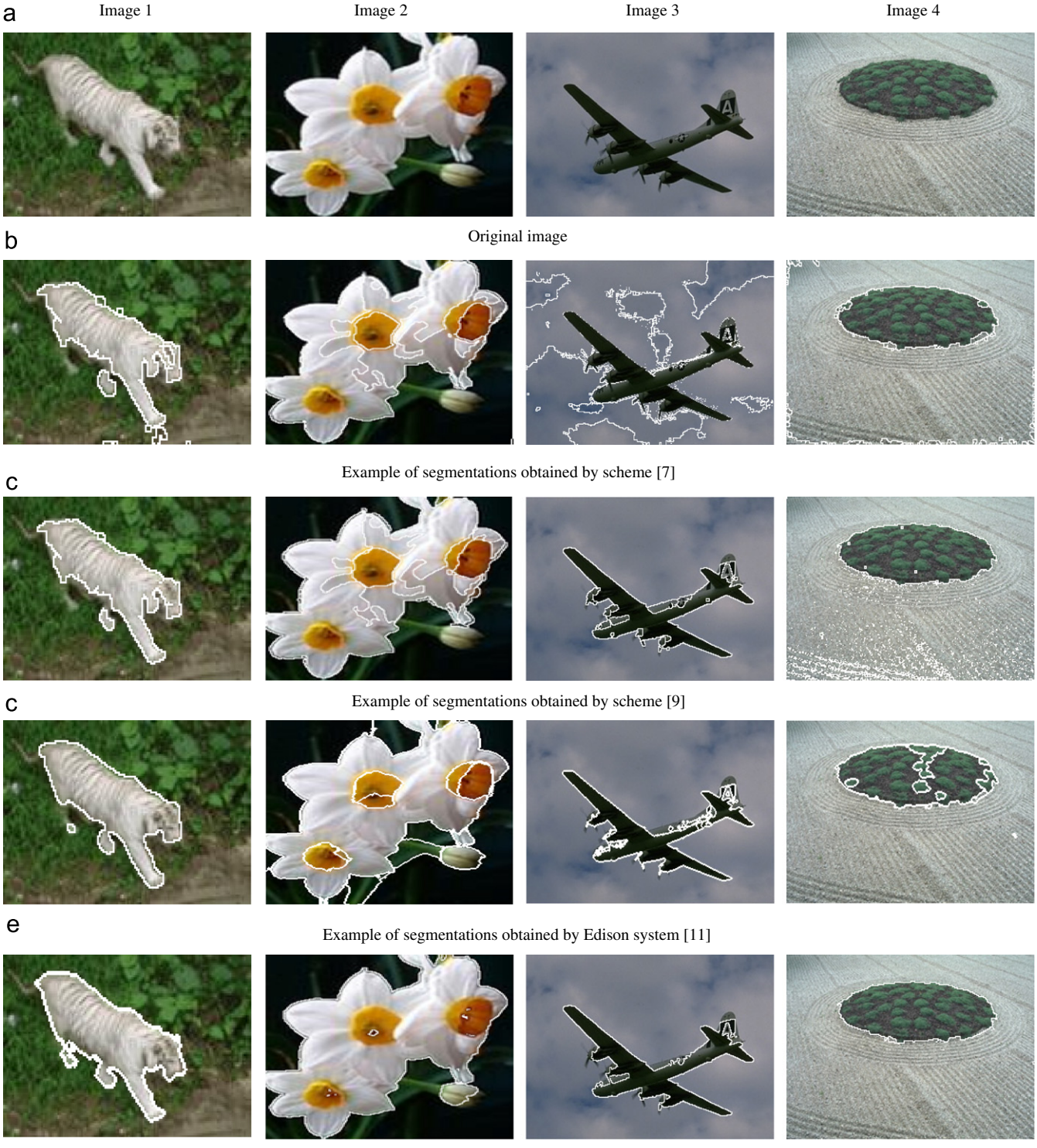

Fig. 6. Segmentation example obtained by four different methods (images 5–8).

accurate classifications than the selection of classification algorithms. For SVM based image segmentation, training pixels can be selected in many ways [16]. A commonly used sampling method is to identify and label small patches of homogeneous pixels in an image. However, adjacent pixels tend to be spatially correlated or have similar values. Training samples collected this way underestimate the spectral variability of each class and are likely to give degraded classification. A simple method to minimize the effect of spatial correlation is random sampling. There are two random sampling strategies, one is called equal sample rate (ESR) in which a fixed percentage of pixels are randomly sampled from each class as training data, and the other is called equal sample size (ESS) in which a fixed number of samples are randomly sampled from each class as training data. In this paper, we will select the training samples for SVM classifiers by using FCM clustering algorithm.

Step 1: Set the initial parameters, such as the number of clusters c, convergence error ε etc.

Step 2: The FCM algorithm is used to cluster the image pixels according to their color and texture features, and the membership value $\mu_k(x_i,y_i)$ $(k=1,2,\ldots,c;\ \ i=1,2,\ldots,n)$ can be obtained. Where n is the number of image pixels.

Step 3: Classify the image pixels by membership value $\mu_k(x_i,y_i)$. Suppose $\mu_J(x_i,y_i)=\max(\mu_1(x_i,y_i),\mu_2(x_i,y_i),\ldots,\mu_c(x_i,y_i))$, then the image pixel at (x_i,y_i) belongs to the Jth cluster.

Step 4: For the image pixels in the Jth cluster, $n_J/10$ image pixels are selected as the training samples of the Jth cluster according to bigger membership value $\mu_J(x_m,y_m)$. Where n_J is the number of image pixels in the Jth cluster. Combine the training samples of all the clusters to form the complete training set. Keep the remaining image pixels as the test set.

(3) *SVM model training*
Train the SVM model (classifier) using the training set created in the previous step.

(4) *SVM pixel classification*
Predict the class labels of the remaining image pixels (test samples) using the trained SVM model (classifier). Combine the training set (class labels given by FCM clustering) and the test set (class labels given by SVM) to obtain the complete label vector and return it as the clustering solution (image segmentation results).

5. Performance evaluation

5.1. Evaluation setup and dataset

Comprehensive experiments were conducted in natural scene images to evaluate the performance of our image segmentation method. The proposed method has been used to segment an image into distinct color-textured regions on the Berkeley segmentation database [1]. This database was selected because it contains hand-labeled segmentations of the images from 30 human subjects. Half of the segmentations involve color images and the other half grayscale images. The database comprises of various images from the Corel dataset and contains ground truth of 300 images for benchmarking image segmentation and boundary detection algorithms. The content of the images are landscapes, animals, portraits and various objects. The proposed algorithm was applied to all 300 images and the output was compared to human perceptual ground truth.

The metrics used for the quantitative evaluation of the proposed algorithm were the following [1,2]:

- The *segmentation error rate* (ER) presents the ratio of misclassified image pixels over the total image pixels, and the error rate is defined as

$$\mathrm{ER}=\frac{N_f+N_m}{N_t}\times 100\%$$

where N_f is the number of false-detection image pixels, N_m denotes the number of miss-detection image pixels, and N_t is total number of images pixels.

- The *local consistency index* (LCI) measures the degree of overlap of the cluster associated with each pixel in one segmentation and its "closest" approximation in the other segmentation. Let S and S' be two segmentations of an image $X=\{x_1,x_2,\ldots,x_N\}$ consisting of N pixels. For a given pixel x_i, consider the classes (segments) that contain x_i in S and S'. $C(S,x_i)$ and $C(S',x_i)$ denote the sets of pixels, respectively. Then, the local refinement error (LRE) is defined at point x_i as

$$\mathrm{LRE}(S,S',x_i)=\frac{|C(S,x_i)\backslash C(S',x_i)|}{|C(S,x_i)|} \tag{20}$$

where $\backslash$ denotes the set differencing operator.

Table 1
The ER, LCI, BCI, and CPU time achieved on the Berkeley segmentation database by four algorithms.

Test image	The quantitative evaluation															
	ER				LCI				BCI				CPU time (s)			
	Scheme [9]	Scheme [7]	Scheme [11]	Our method	Scheme [9]	Scheme [7]	Scheme [11]	Our method	Scheme [9]	Scheme [7]	Scheme [11]	Our method	Scheme [9]	Scheme [7]	Scheme [11]	Our method
Image 1	19.9%	17.4%	20.0%	**13.5%**	0.73	0.81	**0.85**	0.83	0.52	0.56	0.61	**0.67**	195.4	156.5	86.2	**79.2**
Image 2	25.8%	21.8%	29.8%	**12.4%**	0.67	0.72	0.62	**0.81**	0.52	0.63	0.50	**0.68**	129.7	118.6	**51.2**	59.8
Image 3	17.1%	12.2%	20.6%	**9.0%**	0.85	0.90	0.78	**0.94**	0.70	0.75	0.67	**0.81**	107.2	95.6	45.6	**44.3**
Image 4	13.6%	7.6%	7.0%	**4.0%**	0.87	0.89	0.90	**0.92**	0.66	0.68	0.71	**0.74**	130.2	110.5	71.9	**66.9**
Image 5	17.9%	16.8%	16.3%	**14.8%**	0.76	0.79	0.78	**0.80**	0.50	0.51	0.50	**0.53**	192.9	156.5	**74.3**	**78.3**
Image 6	26.2%	20.7%	24.8%	**4.1%**	0.65	0.78	0.59	**0.83**	0.55	0.67	0.53	**0.82**	574.6	498.5	214.8	**193.7**
Image 7	20.8%	16.8%	25.2%	**8.6%**	0.70	0.75	0.67	**0.87**	0.62	0.68	0.61	**0.74**	725.6	689.3	255.5	**243.5**
Image 8	21.7%	17.9%	23.2%	**8.8%**	0.72	0.81	0.69	**0.86**	0.51	0.67	0.49	**0.77**	708.2	684.4	**235.2**	**242.3**

Fig. 7. Comparative results of human-labeled segmentations and the results of different segmentation methods.

Local consistency error (LCE) allows for different directions of refinement in different parts of the image:

$$\mathrm{LCE}(S,S') = \frac{1}{N}\sum_i \min\{\mathrm{LRE}(S,S',x_i),\mathrm{LRE}(S',S,x_i)\}$$

To ease comparison of LCE with measures that quantify similarity between segmentations, LCI = 1 − LCE is defined. The "*I*" in the abbreviations stands for "Index," complying with the popular usage of the term in statistics when quantifying similarity. By implication, LCI lies in the range [0,1] with a value of 0 indicating no similarity and a value of 1 indicating a perfect match.

- The *bidirectional consistency index* (BCI) gives a measure that penalizes dissimilarity between segmentations in proportion to the degree of overlap.
  Consider a set of available ground-truth segmentations $\{S_1,S_2,\ldots,S_K\}$ of an image. The bidirectional consistency error (BCE) measure matches the segment for each pixel in a test segmentation S_{test} to the minimally overlapping segment containing that pixel in any of the ground-truth segmentations.

$$\mathrm{BCE}(S_{test},\{S_k\}) = \frac{1}{N}\sum_{i=1}^{N}\min_k \{\max\{\mathrm{LRE}(S_{test},S_k,x_i),\mathrm{LRE}(S_k,S_{test},x_i)\}\}$$

  However, by using a hard "minimum" operation to compute the measure, the BCE ignores the frequency with which pixel labeling refinements in the test image are reflected in the manual segmentations. To ease comparison of BCE with measures that quantify similarity, the equivalent index BCI = 1 − BCE is defined taking values in [0,1] with a value of 1 indicating a perfect match. The "*I*" in the abbreviations stands for "Index," complying with the popular usage of the term in statistics when quantifying similarity.
- The *CPU time* measures the total run time to segment an image.

5.2. Experimental results

In this section, we demonstrate the segmentation results of the proposed method on natural images obtained from Berkeley segmentation database [1], which also contains the manually segmented benchmark segmentation images. The contrastive experiment results using the normalized cuts image segmentation by Shi and Malik [9], using Edison system by Christoudias et al. [11], and using color- and texture-based image segmentation by Wang and Sun [7] are also presented for comparison. During the experimentation, we found the local window size 5×5 to be most appropriate. The radius-based function (RBF) is selected as the SVM kernel function, and the related parameters settings are: $\gamma=10$, $\sigma^2=0.2$. For other parameter, we set $c=2$, $p=1$, $q=1$.

Figs. 5 and 6 show visual comparison of the proposed color image segmentation algorithm with the scheme [9], system [11], and the method [7]. Table 1 presents the quantitative evaluation (ER, LCI, BCI, and CPU time) for the four different image segmentation algorithms. Looking at the experimental results, it can be seen that the proposed segmentation algorithm performs superior to histogram-based *K*-means clusters method [9], Edison system [11], and the color- and texture-based image segmentation [7] for test images.

Fig. 7 shows the comparative results of human-labeled segmentations and the results of different segmentation methods.

6. Conclusions

Image segmentation is an important low-level preprocessing step for many computer vision problems. In this paper, we have presented a new approach for color image segmentation based on SVM and FCM. Firstly, the pixel-level color feature and texture feature of the image, which is used as input of SVM model (classifier), are extracted via the local homogeneity model and Gabor filter. Then, the SVM model (classifier) is trained by using FCM with the extracted pixel-level features. Finally, the color image is segmented with the trained SVM model (classifier). Results obtained on the Berkeley segmentation database indicate that the proposed algorithm achieves better quantitative results than the state-of-the-art segmentation methods recently proposed in the literature. Drawbacks of the proposed image segmentation are that it lacks enough robustness to noise. Future work will focus on eliminating these drawbacks.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant nos. 60773031 and 60873222, the Open Foundation of State Key Laboratory of Networking and Switching Technology of China under Grant no. SKLNST-2008-1-01, the Open Foundation of State Key Laboratory of Information Security of China under Grant no. 03-06, the Open Foundation of State Key Laboratory for Novel Software Technology of China under Grant no. A200702, and Liaoning Research Project for Institutions of Higher Education of China under Grant nos. 2008351 & L2010230.

References

[1] R. Unnikrishnan, C.E. Pantofaru, M. Hebert, Toward objective evaluation of image segmentation algorithms, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (6) (2007) 929–943.
[2] J.E. Francisco, D.J. Allan, Benchmarking image segmentation algorithms, International Journal of Computer Vision 85 (2) (2009) 167–181.
[3] M. Madhubanti, C. Amitava, A hybrid cooperative–comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding, Expert Systems with Applications 34 (2) (2008) 1341–1350.
[4] M.E. Yuksel, M. Borlu, Accurate segmentation of dermoscopic images by image thresholding based on type-2 fuzzy logic, IEEE Transactions on Fuzzy Systems 17 (4) (2009) 976–982.
[5] X.Y. Zeng, Y.W. Chen, Z. Nakao, H.Q. Lu, Texture representation based on pattern map, Signal Processing 84 (3) (2004) 589–599.
[6] Guan-Yu Chen, Ying-Cheng Chen, et al., Template-based automatic segmentation of drosophila mushroom bodies, Journal of Information Science and Engineering 24 (1) (2008) 99–113.
[7] X. Wang, Y. Sun, A color- and texture-based image segmentation algorithm, Machine Graphics & Vision 19 (1) (2010) 3–18.
[8] R. He, S. Datta, B.R. Sajja, Generalized fuzzy clustering for segmentation of multi-spectral magnetic resonance images, Computerized Medical Imaging and Graphics 32 (5) (2008) 353–366.
[9] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–905.
[10] P. Bao, L. Zhang, X. Wu, Canny edge detection enhancement by scale multiplication, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (9) (2005) 1485–1490.
[11] C. Christoudias, B. Georgescu, P. Meer, Synergism in low-level vision, in: Proceedings of the 16th International Conference on Pattern Recognition, vol. IV, Quebec City, Canada, August 2002, pp. 150–155.
[12] Kuo-Liang Chung, Wei-Jen Yang, Wen-Ming Yan, Efficient edge-preserving algorithm for color contrast enhancement with application to color image segmentation, Journal of Visual Communication and Image Representation 19 (5) (2008) 299–310.
[13] O.G. Ugarriza, L. Saber, E. Vantaram, et al., Automatic image segmentation by dynamic region growth and multiresolution merging, IEEE Transactions on Image Processing 18 (10) (2009) 2275–2288.
[14] Z. Petera, V. Boussone, C. Bergote,g, F. Peyrina., A constrained region growing approach based on watershed for the segmentation of low contrast structures in bone micro-CT images, Pattern Recognition 41 (2008) 2358–2368.

[15] M. Krinidis, I. Pita, Color texture segmentation based on the modal energy of deformable surfaces, IEEE Transactions on Image Processing 18 (7) (2009) 1613–1622.
[16] Xu Haixiang, Cao Wanhua, Chen Wei, Performance evaluation of SVM in image segmentation, in: Proceedings of the Ninth International Conference on Signal Processing (ICSP 2008), Beijing, China, 26–29 October, 2008, pp. 1207–1210.
[17] J.J. Quan, X.B. Wen, Multiscale probabilistic neural network method for SAR image segmentation, Applied Mathematics and Computation 205 (2) (2008) 578–583.
[18] Yuan-Hui Yu, Chin-Chen Chang, Scenery image segmentation using support vector machines, Fundamenta Informaticae 61 (2004) 379–388.
[19] M. Pabitra, B.U. Shankar, K.P. Sankar, Segmentation of multispectral remote sensing images using active support vector machines, Pattern Recognition Letters 25 (9) (2004) 1067–1074.
[20] Jianjun Yan, Jianrong Zheng, One-Class SVM Based Segmentation for SAR Image. Lecture Notes in Computer Science, vol. 4493, 2007, pp. 959–996.
[21] B. Cyganek, Color image segmentation with support vector machines: applications to road signs detection, International Journal of Neural Systems 18 (4) (2008) 339–345.
[22] Jih-Jeng Huang, Gwo-Hshiung Tzeng, Chorng-Shyong Ong, Marketing segmentation using support vector clustering, Expert Systems with Applications 32 (2) (2007) 313–317.
[23] Jun Ji, Fengjing Shao, Rencheng Sun, A TSVM based semi-supervised approach to SAR image segmentation, in: Proceedings of the 2008 International Workshop on Education Technology and Training & 2008 International Workshop on Geoscience and Remote Sensing, vol. 1, Shanghai, China, 2008, pp. 495–498.
[24] Z. Xue, L. Long, S. Antani, G.R. Thoma, J. Jeronimo, Segmentation of mosaicism in cervicographic images using support vector machines, Proceedings of SPIE Medical Imaging 7259 (1) (2009) 72594X–72594X-10.
[25] B. Schölkopf, A.J. Smola, in: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT Press, Cambridge, MA, 2002.
[26] V. Vapinik, in: The Nature of Statistical Learning Theory, Springer-Verlag, New York, 2000.
[27] C. Liu., Gabor-based Kernel PCA with fractional power polynomial models for face recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (5) (2004) 572–581.

Xiangyang Wang was born in Tieling, China, in 1965. He is currently a professor with the School of Computer and Information Technology at the Liaoning Normal University, China. He obtained his B.S. degree from the Lanzhou University, China and his M.S. degree from the Jilin University, China, in 1988 and 1995, respectively. His research interests include signal processing and communications, digital multimedia data hiding and information assurance, applications of digital image processing, computer vision. He has published more than 150 journal papers, 20 conference papers, and contributed in 2 books in his areas of interest.

Ting Wang received the B.S. degree from the School of Computer and Information Technology, Liaoning Normal University, China, in 2008, where she is currently pursuing the M.S. degree. Her research interests include signal processing and image segmentation.

Juan Bu received the B.S. degree from the School of Computer and Information Technology, Liaoning Normal University, China, in 2007, where she is currently pursuing the M.S. degree. Her research interests include signal processing and image segmentation.