# Color Image Segmentation Using SVM Pixel Classification Image

K. Sakthivel, R. Nallusamy, C. Kavitha

***Abstract*—**The goal of image segmentation is to cluster pixels into salient image regions. Segmentation could be used for object recognition, occlusion boundary estimation within motion or stereo systems, image compression, image editing, or image database look-up. In this paper, we present a color image segmentation using support vector machine (SVM) pixel classification. Firstly, the pixel level color and texture features of the image are extracted and they are used as input to the SVM classifier. These features are extracted using the homogeneity model and Gabor Filter. With the extracted pixel level features, the SVM Classifier is trained by using FCM (Fuzzy C-Means).The image segmentation takes the advantage of both the pixel level information of the image and also the ability of the SVM Classifier. The Experiments show that the proposed method has a very good segmentation result and a better efficiency, increases the quality of the image segmentation compared with the other segmentation methods proposed in the literature.

***Keywords*—**Image Segmentation, Support Vector Machine, Fuzzy C–Means, Pixel Feature, Texture Feature, Homogeneity model, Gabor Filter.

## I. INTRODUCTION

THE people are only interested in certain parts of the image in the research and application of the image. These parts are frequently referred as a target or foreground (other part is called background), they generally correspond to the image in a specific and unique nature of the area. It needs to extract and separate them in order to identify and analyze object, on this basis it will be possible to further use for the target. Image segmentation is a technique and process which divide the image into different feature of region and extract out the interested target. Here features can be pixel grayscale, color, texture, etc, Image segmentation plays a fundamental role in many computer vision applications.

In pattern recognition approaches image segmentation enables the isolation of single objects or their parts in the scene that can be subsequently identified in an easier and more accurate way [1]. Image segmentation has 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: thresholding, template matching, clustering, edge detection and region growing. These algorithms have been 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.

Recently, intelligent approaches, such as neural network and support vector machine (SVM) [5], have already been utilized successfully in image segmentation. Quan and Wen [6] 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

Dr. K. Sakthivel, Professor, is with the Department of Computer Science and Engineering, K. S. Rangasamy College of Technology, Tiruchengode, Namakkal-637215, Tamilnadu, India (phone: +91-9942166425; e-mail: k_sakthivel72 @yahoo.com).

Dr. R. Nallusamy, Principal, is with the Nandha College of Technology, Erode, TamilNadu, India (e-mail: nallsam@rediffmail.com).

Dr. C.Kavitha, Assistant Professor, is with the Department of Computer Science, Thiruvallur Government Arts College, Rasipuram, TamilNadu, India (e-mail: kavithachellappan@yahoo.com).

strongpoint from both of them, is designed. Yu and Chang [8] presented an effective and efficient method for solving scenery image segmentation by applying the SVMs methodology. In scheme [10], 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 [11] 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 [12] 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 [13], 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. 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 [14]. The 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 [15]. 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 support vector machine (SVM) pixel classification. At first, the pixel level color and texture features of the image are extracted and they are used as input to the SVM classifier. These features are extracted using the homogeneity model and Gabor Filter. With the extracted pixel level features, the SVM Classifier is trained by using FCM (Fuzzy C-Means). Finally, the color image is segmented with the trained SVM classifier. The image segmentation takes the advantage of both the pixel level information of the image and also the ability of the SVM Classifier. Experimental results show that the proposed method achieves competitive segmentation results compared to the state-of-the-art segmentation methods recently proposed in the literature.

## II. Support Vector Machine (SVM)

Support vector machines (SVMs) are a set of related supervised learning methods used for classification and regression [16]. They belong to a family of generalized linear classifiers. In another terms, Support Vector Machine (SVM) is a classification and regression prediction tool that uses machine learning theory to maximize predictive accuracy while automatically avoiding over-fit to the data. Support Vector Machines can be defined as systems which use hypothesis space of a linear functions in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias derived from statistical learning theory The foundations of Support Vector Machines (SVM) have been developed by Vapnik [2] and gained popularity due to many promising features such as better empirical performance. 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.

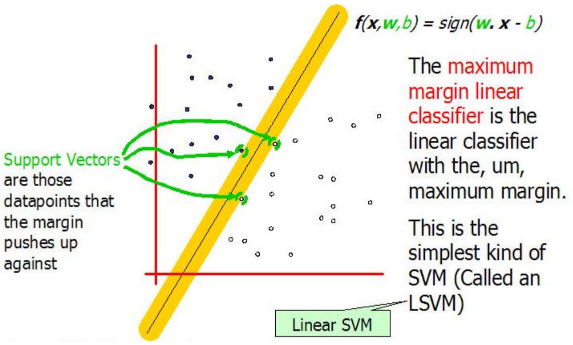
f(x,w,b) = sign(w. x - b)
The maximum margin linear classifier is the linear classifier with the, um, maximum margin.
This is the simplest kind of SVM (Called an LSVM)
Support Vectors are those datapoints that the margin pushes up against
Linear SVM

Fig. 1 Illustration of SVM (Linear)

Expression for Maximum margin is given as

$$m\arg in \equiv \arg\min_{X\in D} d(x) = \arg\min_{X\in D} \frac{|X.W+b|}{\sqrt{\sum_{i=1}^{d} w_i^2}} \tag{1}$$

### A. SVM Representation

In this we present the QP formulation for SVM classification. This is a simple representation only.

1. SV classification

$$\min_{f,\xi_i} \| f \|_K^2 + C\sum_{i=1}^{l} \xi_i \; y_i f(x_i) \geq 1-\xi_i, \, for\, all\, i\, \xi_i \geq 0 \tag{2}$$

2. SVM classification, Dual formulation:

$$\min_{\alpha_i} \sum_{i=1}^{1} \alpha_i - \frac{1}{2} \sum_{i=1}^{1}\sum_{j=1}^{1} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$
$$\sum_{i=1}^{l} \alpha_i y_i = 0 \tag{3}$$

$0 \leq \alpha_i \leq C$, for all $i$;

Variables $\xi_i$ are called slack variables and they measure the error made at point (xi,yi). Training SVM becomes quite challenging when the number of training points is large.

### B. Soft Margin Classifier

In real world problem it is not likely to get an exactly separate line dividing the data within the space. And we might have a curved decision boundary. We might have a hyper plane which might exactly separate the data but this may not be desirable if the data has noise in it. It is better for the smooth boundary to ignore few data points than be curved or go in loops, around the outliers. This is handled in a different way; here we hear the term slack variables being introduced. Now we have, yi(w'x + b) ≥ 1 - Sk. This allows a point to be a small distance Sk on the wrong side of the hyper plane without violating the constraint. Now we might end up having huge slack variables which allow any line to separate the data, thus in such scenarios we have the Lagrangian variable introduced which penalizes the large slacks.

$$\min L = \frac{1}{2}W'W - \sum \lambda_k (y_k (W'X_k + b) + s_k - 1) + \alpha \sum s_k \quad (4)$$

where reducing α allows more data to lie on the wrong side of hyper plane and would be treated as outliers which give smoother decision boundary.

## III. 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.

### *A. Pixel Color Feature*

Color is one of the most dominant and distinguishable low-we introduce a new pixel-level color feature according to local homogeneity model. 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 does not.

The pixel-level color feature of color component Pkij (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 dxd (for example, 5x5) window centered at (i,j) for the computation of pixel-level color feature, and d is odd integer greater than 1.

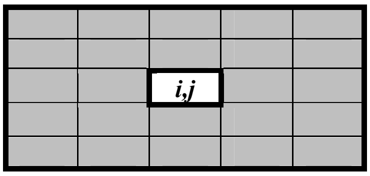

Fig. 2 Local Image Window

2. Compute the Pixel Color Feature of Component $P^{k}_{ij}$

Homogeneity is largely related to the local information extracted from an image and reflects how uniform a region is [4]. 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 Pkij.

The standard deviation of color component Pkij (k=L,a,b) calculated as

$$v_{ij}^{\ k} = \sqrt{(1/d^2 \sum_{m=i-(d-\frac{1}{2})}^{i+(\frac{d-1}{2})} \sum_{n=j-(\frac{d-1}{2})}^{j+(\frac{d-1}{2})} (p^{k}_{\ mn} - \mu^{k}_{\ ij})^2)} \quad (5)$$
$$where\, 0 \le i, m \le M-1, 0 \le j, n \le N-1$$

$\mu^{k}_{\ ij}$ is the mean of color component Pkij (k=L,a,b) within window wij and calculated as

$$\mu^{k}_{\ ij} = 1/d^2 \sum_{m=i-(d-\frac{1}{2})}^{i+(\frac{d-1}{2})} \sum_{n=j-(\frac{d-1}{2})}^{j+(\frac{d-1}{2})} p^{k} mn \quad (6)$$

The discontinuity for color component Pkij 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 ekij (k=L,a,b ) of the gradient at location (i,j) as the measurement:

$$e_{ij}^{\ k} = \sqrt{(G^{k^2}_{\ x} + G^{k^2}_{\ y})} \quad (7)$$

where Gxk and Gxk are the components of the gradient of color component Pkij (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:

$$V^{k}_{ij} = V^{k}_{ij} / V^{k}_{\max}, E^{k}_{ij} = e^{k}_{ij} / e^{k}_{\max} \quad (8)$$

where

$$\begin{gathered} V^{k}_{\max} = \max\{V^{k}_{ij}\}, e^{k}_{\max} = \max\{e^{k}_{ij}\}, \\ (0 \le i \le M-1, 0 \le j \le N-1), (k = L, a, b) \end{gathered} \quad (9)$$

The local homogeneity is represented as

$$\begin{gathered} CF^{k}_{ij} = H^{k}_{ij} = 1 - E^{k}_{ij} \times V^{k}_{ij} \\ where (0 \le i \le M-1, 0 \le j \le N-1), (k = L, a, b) \end{gathered} \quad (10)$$

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, operations are 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 5x5 window for computing the standard deviation and discontinuity.

So, we can obtain the pixel-level color feature CFij of the image pixel Pij at location (i,j)

$$CF_{ij} = (H_{ij}^{L}, H_{ij}^{a}, H_{ij}^{b}) \tag{11}$$

*B. 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 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 sub bands 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:

$$\begin{aligned} g(x,y) &= h(x,y)\exp(2\pi jWx) \\ &= {}^{1}\!/_{2}\,\pi\sigma_x\sigma_y \exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma^2{}_x}+\frac{y^2}{\sigma^2{}_y}\right)\right]\exp(2\pi jWx) \end{aligned} \tag{12}$$

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

$$\begin{aligned} G(u,v) &= H(u-W,v) \\ &= \exp\{-\frac{1}{2}[(u-W)^2/\sigma_u^2 + v^2/\sigma_v^2]\} \end{aligned} \tag{13}$$

Here h(x,y) is a two dimensional Gaussian function with its center at the origin, and $\sigma_x$ and $\sigma_y$ denote its variances in x and y directions, respectively. The variances of the Fourier transform function G (*u,v*) are and

$$\sigma_v = 1/2\pi\sigma_y \tag{14}$$

$\sigma$ 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:

$$\begin{gathered} g_{mn}(x,y) = a^{-2m} g(x', y'), a \geq 1 \\ \begin{pmatrix} x' \\ y' \end{pmatrix} = a^{-m} \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \theta = \frac{n\pi}{L}; m = 0,1,\ldots K-1; n = 0,1,\ldots L-1 \end{gathered} \tag{15}$$

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:

$$W = 1, a = 2, \sigma_x = \sigma_y = \frac{1}{2\pi}, K = 3, L = 4$$

Let I(x,y) denote an image with size w xh. The Gabor-filtered output Gmn(x,y) of the image I(x,y) is defined as its convolution with the Gabor filter Gmn (x,y):

$$\begin{aligned} G_{mn}(x,y) &= l(x,y) * g_{mn}(x,y) \\ &= \sum\nolimits_{x_1=1}^{w} \sum\nolimits_{y_1=1}^{h} l(x-x_1, y-y_1) g_{mn}(x_1,y_1) \end{aligned} \tag{16}$$

Here, Gmn(x,y) is the Gabor-filtered image at the scale parameter m and the orientation parameter n.

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 sub band coefficients. Various nonlinear operations have been used to boost up the sparse sub band coefficients. Our approach is based on the local median energy of the sub band 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 3 orientation and 3 scale sub bands. We use the 3 orientation and 3 scale bands. Our goal is to identify regions with the dominant orientation (0 , 45 , 90 ) and it is the maximum of the 9 coefficients that determines the orientation at a given pixel location.

## IV. Image Segmentation Using SVM and FCM

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 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 shown in Fig. 3.

### *A. Pixel-Level Color and Texture Features Extraction*

The pixel-level color and texture features are extracted via the local homogeneity model and Gabor filter.

### *B. 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 accurate classifications than the selection of classification algorithms. 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 e etc.

Step 2.The FCM algorithm is used to cluster the image pixels according to their color and texture features, and the membership value µk(xi,yi) (k = 1,2,. . .,c; i= 1,2,. . .,n) can be obtained. Where n is the number of image pixels.

Step 3.Classify the image pixels by membership value µk(xi,yi).Suppose

$$\mu_J(x_i, y_i) = \max(\mu_1(x_i, y_i), \mu_2(x_i, y_i), ....\mu_c(x_i, y_i)),$$

then the image pixel at (xi,yi) belongs to the Jth cluster.

Step 4.For the image pixels in the Jth cluster, nJ=10 image pixels are selected as the training samples of the Jth cluster according to bigger membership value µJ(xm,ym). Where nJ is the numbe rof 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.

### *C. SVM Model Training*

Train the SVM classifier using the training set created in the previous step.

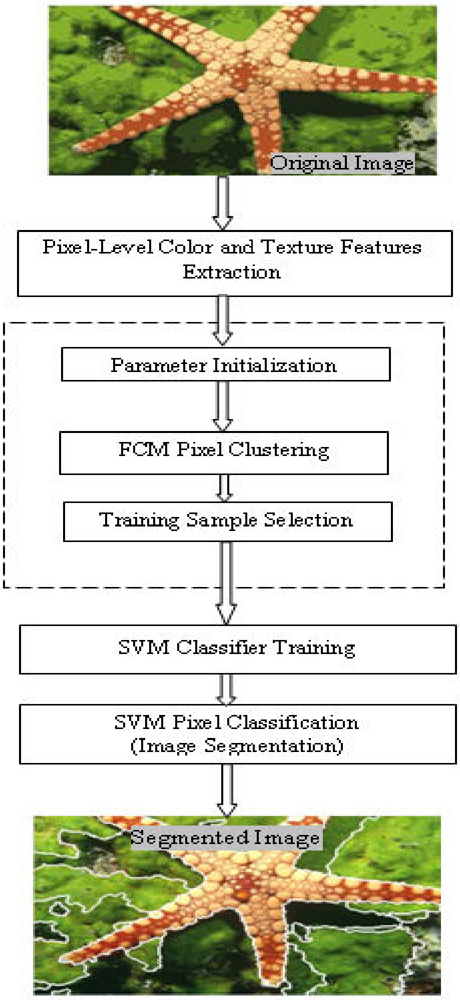
Original Image
Pixel-Level Color and Texture Features Extraction
Parameter Initialization
FCM Pixel Clustering
Training Sample Selection
SVM Classifier Training
SVM Pixel Classification (Image Segmentation)
Segmented Image

Fig. 3 Image Segmentation using SVM Pixel Classification

### *D. SVM Pixel Classification*

Predict the class labels of the remaining image pixels using the trained SVM 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)

## V. Performance Evaluations

### *A. Evaluation Setup*

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. This database was selected because it contains hand-labeled segmentations of the images from 20 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 200 images for benchmarking image segmentation and boundary detection algorithms. The proposed algorithm was applied to all 200 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:

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

$$ER = \frac{N_f + N_m}{N_t} \times 100\% \tag{17}$$

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. 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 = {$x_1$ ,$x_2$…..$x_N$} 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

$$LRE(S,S',x_i) = |C(S,x_i) \setminus C(S',x_i)| / C(S,x_i) \tag{18}$$

where \ denotes the set differencing operator.

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

$$LCE(S,S') = 1/N \sum_i \min\{LRE(S,S',x_i), LRE(S',S,x_i)\} \tag{19}$$

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 {S1,S2,…..,Sk}of an image. The bidirectional consistency error (BCE) measure matches the segment for each pixel in test segmentation Stest to the minimally overlapping segment containing that pixel in any of the ground-truth segmentations.

$$\begin{aligned} &BCE(S_t\{S_k\}) \\ &= 1/N \sum_{i=1}^{N} \min\{\max\{LRE(S_{test},S_k,x_i), LRE(S_k,S_{test},x_i)\}\} \end{aligned} \tag{20}$$

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.

### *B. Experimental Results*

In this section, we demonstrate the segmentation results of the proposed method on natural images obtained from Berkeley segmentation database, which also contains the manually segmented benchmark segmentation images [3]. The contrastive experiment results using the normalized cuts image segmentation by [9] and using color- and texture-based image segmentation by [7] are also presented for comparison. During the experimentation, we found the local window size 5x5 to be most appropriate. The radius-based function (RBF) is selected as the SVM kernel function, and the related parameters settings are: γ=10, $\sigma^2 =$ 0.25. For other parameter, we set c=2.5, p=1, q=1.

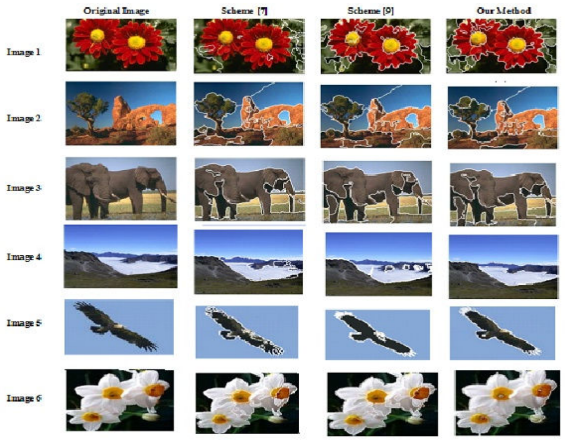

Fig. 4 Segmentation Obtained by Three Different methods

Fig. 4 shows visual comparison of the proposed color image segmentation algorithm with the scheme [7], and the scheme [9]. Tables I-III present the quantitative evaluation of ER, LCI and BCI respectively for the three 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], and the color- and texture-based image segmentation [7] for test images.

TABLE I
THE QUANTITATIVE EVALUATION - ER

| Test Image | Scheme[7] | Scheme[9] | Our Method |
|---|---|---|---|
| **Image1** | 17.40% | 19.90% | **13.75%** |
| **Image2** | 21.80% | 25.80% | **12.80%** |
| **Image3** | 12.20% | 17.10% | **8.75%** |
| **Image4** | 7.60% | 13.60% | **5.00%** |
| **Image5** | 16.80% | 17.90% | **15.00%** |
| **Image6** | 20.70% | 26.20% | **4.00%** |

TABLE III
THE QUANTITATIVE EVALUATION - LCI

| Test Image | Scheme[7] | Scheme[9] | Our Method |
|---|---|---|---|
| **Image1** | 0.81 | 0.73 | **0.85** |
| **Image2** | 0.72 | 0.67 | **0.75** |
| **Image3** | 0.9 | 0.85 | **0.90** |
| **Image4** | 0.89 | 0.87 | **0.90** |
| **Image5** | 0.79 | 0.76 | **0.82** |
| **Image6** | 0.78 | 0.65 | **0.85** |

TABLE IIIII
THE QUANTITATIVE EVALUATION - BCI

| Test Image | Scheme[7] | Scheme[9] | Our Method |
|---|---|---|---|
| **Image1** | 0.56 | 0.52 | **0.65** |
| **Image2** | 0.63 | 0.52 | **0.70** |
| **Image3** | 0.75 | 0.7 | **0.80** |
| **Image4** | 0.68 | 0.66 | **0.75** |
| **Image5** | 0.51 | 0.5 | **0.55** |
| **Image6** | 0.67 | 0.55 | **0.85** |

## VI. CONCLUSION

In this paper, we have presented a new approach for color image segmentation based on SVM and FCM. First, the pixel-level color feature and texture are extracted via the homogeneity model and Gabor filter. They are used as input to the SVM Classifier. Then, the SVM Classifier is trained by using FCM with the extracted pixel-level features. Finally, the color image is segmented with the trained SVM model. Results obtained on the Berkeley segmentation database indicate that the proposed algorithm achieves better quantitative results than the 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.

## REFERENCES

[1] X. Haixiang, C. Wanhua, C. Wei, C and G. Liyuan, "Performance evaluation of SVM in image segmentation", *In 2008 9th International Conference on Signal Processing*, pp. 1207-1210, 2008.

[2] V. Vapnik, "The nature of statistical learning theory", springer., New York, 2000.

[3] J.E. Francisco, D.J and Allan, "Benchmarking image segmentation algorithms", *International Journal of Computer Vision*, vol.85, no.2, pp.167–181, 2009.

[4] R. Unnikrishnan, C.E. Pantofaru and M. Hebert, "Toward objective evaluation of image segmentation algorithms", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.29, no.6, pp. 929–943, 2007.

[5] M. Song and D. Civco, "Road extraction using SVM and image segmentation", *Photogrammetric Engineering & Remote Sensing*, vol. 70, no. 12, pp. 1365-1371, 2004.

[6] J.J. Quan and X.B. Wen, "Multiscale probabilistic neural network method for SAR image segmentation", *Applied Mathematics and Computation*,vol. 205, no.2,,pp.578–583, 2008.

[7] X. Wang and Y. Sun, "A color- and texture-based image segmentation algorithm", *Machine Graphics & Vision*, vol.19, no.1, pp. 3–18, 2010.

[8] A. H. Yu, and C.C. Chang, "Scenery image segmentation using support vector machines", *Fundamenta Informaticae*, vol. 61, no. 3, pp. 379-388, 2004.

[9] J. Shi and J. Malik, "Normalized cuts and image segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no.8, pp.888–905, 2000.

[10] M. Pabitra, B.U. Shankar and K.P. Sankar, "Segmentation of multispectral remote sensing images using active support vector machines", *Pattern Recognition Letters*, vol.25, no.9, pp. 1067–1074, 2004.

[11] J, Yan and J. Zheng, "One-class SVM based segmentation for SAR image", *In Advances in Neural Networks*–ISNN 2007, pp. 959–996, 2007.

[12] B. Cyganek, "Color image segmentation with support vector machines:applications to road signs detection", *International Journal of Neural Systems*, vol.18, no.4, pp. 339–345, 2008.

[13] J.J. Huang, G.H. Tzeng and C. S. Ong, "Marketing segmentation using support vector clustering", "*Expert systems with applications*", vol. 32, no. 2, pp. 313-317, 2007.

[14] J. Ji, F. Shao, R. Sun, N. Zhang and G. Liu, "A TSVM based semi-supervised approach to SAR image segmentation". *In Education Technology and Training, 2008. and 2008 International Workshop on Geoscience and Remote Sensing*, International Workshop, IEEE, Vol. 1, pp. 495-498, 2008.

[15] Z. Xue, L. Long, S. Antani, G.R. Thoma and J. Jeronimo, "Segmentation of mosaicism in cervicographic images using support vector machines" *Proc SPIE Med Imaging,* vol. 7259, no.1 pp. 72594X–72594X-10, 2009.

[16] B. Boser, I. Guyon, and V. Vapnik, "An training algorithm for optimal margin classifiers", *In Proceedings of the Fifth Annual Workshop on Computational Learning Theory,* pp. 144–152,1992.