Color Image Segmentation Using Hybrid Learning Techniques

C. Mala* and M. Sridevi
Department of Computer Science and Engineering
National Institute of Technology
Tiruchirappalli, India - 620 015.
{mala, msridevi}@nitt.edu

Abstract

Image segmentation is the process of finding out all non-overlapping distinct regions from the given image based on certain criteria such as intensity, color, texture or shape. This paper proposes a two level hybrid non classical model for image segmentation based on pixel color and texture features of the image. The first level uses Fuzzy C-Means (FCM) unsupervised method to form a clustering among all the pixels based on color and texture properties. And the second level uses supervised methods [Classification Tree and Adaboost] for color learning and pixel classification, with the prior knowledge of the image obtained from the FCM. Experiment was conducted for the three different supervised classification methods including Support Vector Machine (SVM) and their performances are analysed. From the results, it is inferred that Adaboost classifier increases accuracy and reduces misclassification error when compared to Classification Tree and SVM methods.

Keywords: support vector machine, adaboost, classification tree, fuzzy c-mean clustering, texture, color space.

1 Introduction

Image acquisition, enhancement, restoration, compression, segmentation, recognition are the basic steps involved in digital image processing. Image processing plays a vital role in many applications such as medical, biometric, health care, transportation, recognition, entertainments and games, mining, retrieval, military, etc. Based on the recent improvement in digital communication of text, audio and video, convergence provides faster retrieval and transfer of information of an image for analysis and processing. It acts as an interlink of computing and other information technologies, media content, and communication networks that has arisen as the result of the evolution and popularization of the internet as well as the activities, products and services that have emerged in the digital media space. It is a combination of a traditional and an innovative technique that can be used in any product related to analysis of media content, a system or process that demonstrates its capability in media convergence. This type of technique can be increasingly carried out in digital media spaces across a growing network of information and communication technology devices.

The main objective of image segmentation is to partition the given image into various regions with respect to various criteria such as intensity, color or texture. It is a process of grouping each pixel by sharing some visual characteristics. It is a low-level vision process and most widely used in image processing, image analysis, image understanding and computer vision. It is a crucial step in video and vision applications, like object and pattern recognition, localization, classification, data compression, tracking, image retrieval and understanding. Segmentation results will affect all the successive processes of image analysis, such as feature measurement, object representation and description and even the following

IT CoNvergence PRActice (INPRA), volume: 2, number: 2, pp. 21-42
*Corresponding author: Tel: +91-(0)431-2503208

higher level tasks such as object classification and scene interpretation as it is one of the most critical tasks in automatic image analysis [7, 25].

In the past three decades, a variety of image segmentation methods have been developed for image processing and computer vision applications. The algorithms can be applied to gray scale [12] and color images [8, 22, 31]. These algorithms can be broadly classified into six categories namely: Edge detection, Region based segmentation, Thresholding, Template matching, Clustering and Supervised Learning. Edge detection [6, 23, 28, 29] is based on the detection of edge points and focuses on images with sharp edges. Region growing method [18, 19, 25] works with spatial partitioning, its processing is sequential and it concentrates on the order in which pixels are processed. Thresholding [7,24] is a simple and efficient algorithm and it is best suited for bimodal distribution of images. Template matching [31] performs matching of a template in an image to find region of interest. Clustering [14,22] uses two major algorithms such as divisive and agglomerative clustering. Both algorithms deal with lot of pixels to form regions. Supervised learning method such as Support Vector Machine (SVM) [4, 5, 8, 31] and Neural Network (NN) [20, 22] are used for segmentation problems. The accuracy of the method depends on the selected features. The successive high level processes such as image analysis and understanding is influenced by segmented results. Therefore, it is important to segment the image into meaningful regions in an accurate, fast, automated and robust manner. However, it is a difficult process if the image is highly ill-posed.

In reality the variation in color and texture are dominant in color images. This content of the image can be used to decide or identify the region of interest in any given image. Hence segmentation of color images helps in improving the accuracy and efficiency of detecting the desired region of interest. Image segmentation performed based on texture yields better results when compared to color property. They are useful to detect objects such as land, road, sea, mountain, forest areas from the satellite images, blood cells and tissues in medical images, to inspect electric components and to control traffic automatically, etc. Texture segmentation is a challenging process to distinguish and classify the given texture. Indeed, the main bottleneck to segment a texture image is to find an appropriate set of generic computable descriptors to characterize a given texture and discriminate distinct textures between them. The texture representation and characterization is also an important open problem as there is no consensus on how to define a texture model, despite several attempts including random models in spatial or transform domains, low-dimensional manifold models, or sparsity-oriented model . There are number of papers devoted to texture segmentation of images [14]. In recent times, image segmentation using color and texture descriptors [13, 31] are applied for segmentation because both are dependent image characteristics. The blend of color image and hybrid technique leads in developing new technologies that are deviated from the traditional methods. This improved method accelerates in development of convergence technologies such as data mining, image processing etc and also acts as a bridge between information retrieval and image processing technologies. Due to the increase in the usage of digital media applications, digital convergence is needed. Convergence provides better understanding on visual contents. Image segmentation is done to aid digital convergence.

2 Related Works

Edge-based image segmentation methods [6,11,12,28,29] perform segmentation based on abrupt changes in intensity values of the image. Sobel, Prewitt, Laplacian operators and Canny are some of the edge detection algorithms. The edge-based methods are insensitive to non - stationary image and noise.

Region-based image segmentation methods [18,19,25] perform the task based on homogeneity property such as color, intensity, texture or shape information. The initial seed determination and stop criteria are main challenges in this technique.

Another popularly used technique is thresholding [7,10,24]. In this, the segmentation result depends on the selection of threshold value. Histogram based thresholding is a commonly used thresholding method due to its simplicity and efficiency. This technique is best suited for bimodal images where the objects reside upon a contrast background. They do not perform well for unimodal [12], when the region is much smaller than the background area. The segmented result may lead to pseudo or missing of edge information.

Graph based methods can be adopted for image segmentation problems [3,17,21]. In these methods, nodes indicate pixel values and edges indicate connection between neighboring pixels. Graph-based segmentation approaches suffer from high computational complexity.

The image segmentation is a type of classification problem which groups the pixels based on similarity or dissimilarity properties of the image. Hence learning methods can be adopted for image segmentation problems. The unsupervised learning methods such as K means and Fuzzy C-Means clustering algorithms are successfully applied to segmentation problems in [14, 22, 30]. They are iterative techniques used to partition an image into K clusters. Each pixel is assigned to any one of the clusters based on the minimum distance between the pixel and the cluster center. The major drawbacks of these methods are: i) They may not provide optimal solution. ii) The output of segmentation mainly depends on the initial cluster size and centeriod values of each cluster.

More recently, supervised intelligent approaches such as SVM [1,5,8,15,26,31] and Neural network [20,22,34] have already been used for image segmentation. They provide successful segmented results and have been used in many applications such as road signs recognition system [8], automatic pipe inspection [12], contour detection of breast tumors [4], Skin color segmentation [15], etc. In [34], cancer detection [11], face detection [32], brain lesion segmentation [33], a new modified SVM method is adopted for the segmentation problem and pruning strategy was used to preserve support vectors while eliminating redundant training vectors. This is a complex technique. Based on parameter setting, the results of segmentation may be over or under segmented, which leads to undesired results.

Hybrid model [15, 31] using the Support Vector Machine (SVM) and Fuzzy C-Means (FCM) clustering is another solution for image segmentation problem. Both local and spatial similarity measure model based color features and Steerable filter based texture features are extracted from the image. FCM based algorithm is employed for clustering color and texture features of the image. The features are fed to SVM algorithm for classification of various regions. In [31], pixel wise support vector machine classification is adopted for color image segmentation. The SVM classifier accepts the pixel-level color feature and texture feature as input and they are extracted using local homogeneity model and Gabor filter respectively. Then image is segmented by trained classifier using FCM. The samples for training are attained by FCM clustering algorithm. This segmentation utilizes the local information of the image. However this method cannot handle noisy image.

In [15], a skin color image segmentation using Fuzzy System learned through Fuzzy Clustering and Support Vector Machine (FS-FCSVM) is proposed. The FS-FCSVM is a fuzzy system constructed by fuzzy if-then rules with fuzzy singletons in the consequence. From the experimental results, it is inferred that these methods revealed good segmentations. However, the approach suffers from the high computational cost.

A new approach based on neural network [22] was developed for image segmentation problems. An Adaptive Probabilistic Neural Network (APNN) and level set method are used for brain segmentation of Magnetic Resonance Imaging (MRI). The APNN is employed to classify the input MR image, and to extract the initial contours. Based on the extracted contours as the initial zero level set contours, the modified level set evolution are performed to accomplish the segmentation.

In [20], an effective multi- scale method for segmenting Synthetic Aperture Radar (SAR) images using Probabilistic Neural Network (PNN) was proposed. By combining PNN with Multi-scale Auto - Regressive (MAR) model, a classifier is designed. It provides better segmentation results. Nevertheless, this approach is computationally expensive.

Most of the sophisticated techniques [23, 24, 28] may not provide a satisfactory segmentation for complex images. The local information of the image is utilized for color based image segmentation process. But it cannot succeed in optimal segmentation for complex images since the features taken as input are not appropriate for complex images in an effective manner. During segmentation process, diverse textures are found in the image. So this method cannot handle noisy and textured images. FCM algorithm usually performs well with noise-free images, but gives poor results if the images are corrupted by noise, outliers, and other imaging artifacts. Integrating color and texture features [13, 31] for unsupervised segmentation improves the segmented result when compared to either using color or texture feature.

SVM-based image segmentation methods achieve better segmentation due to the generalization ability of SVM classifier [26], but they are sensitive to pixel-level feature and initialization.

To overcome the above mentioned drawbacks, this paper proposes a hybrid model using FCM and Supervised learner for image segmentation problem based on pixel level color and texture features. The model compares three classifiers such as Classification Tree, SVM, Adaboost with FCM to provide reduced misclassification error.

The remaining of the paper is organized as follows: Section 3 explains Pixel level Segmentation for three different classifiers such as Classification Tree, SVM, Adaboost algorithms. The experimental results and their performance analysis are discussed in Section 4. Finally, Section 5 concludes the paper.

3 Pixel Level Segmentation based on Various Classifiers

The aim of segmentation is to partition the image into homogenous regions based upon pixel properties. The segmentation is a classification problem where each and every pixel is classified into different groups. The effective features are extracted based on pixel level and a label is assigned to every pixel through which regions are formed. The flow diagram of pixel level image segmentation process is shown in Fig. 1. The color transformation is applied to the input color image. The Pixel level and texture features are selected from the transformed color image. The feature vector is created for the selected features. The following unsupervised and supervised machine learning methods are applied to the selected features to determine the segmented regions:

1. FCM.
2. Classification Tree.
3. SVM.
4. Adaboost algorithm.

3.1 Color Transformation

The color space such as HSV (Hue,Saturation,Value), YUV, YIQ [11] can be used for image segmentation purpose. But many of them are not uniform. Therefore the input color image has to be transformed into CIE $L^*a^*b^*$ color space as it is the best model when compared to other color models [31]. The minor color difference can be measured very easily in CIE $L^*a^*b^*$ space. To extract pixel level color and texture features, LAB color space is applied to the original image where L is Luminosity or brightness layer, A is in chromaticity layer where color falls along the Red Green axis and B is also in chromaticity layer where color falls along Blue Yellow axis.

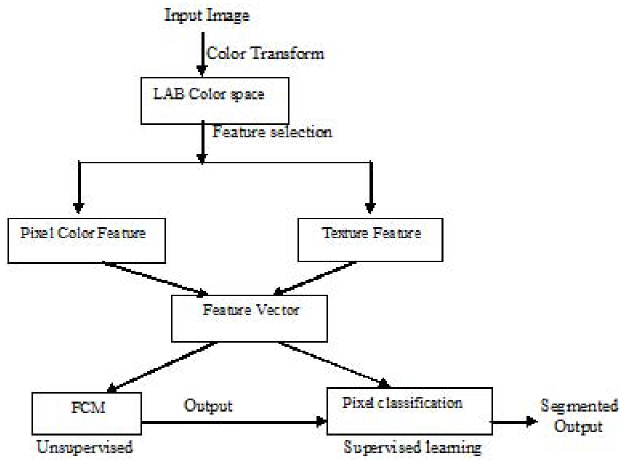


Figure 1: Flow diagram of Pixel level segmentation.

$(i-1, j-1)$	$(i-1, j)$	$(i-1, j+1)$
$(i, j-1)$	(i, j)	$(i, j+1)$
$(i+1, j+1)$	$(i+1, j)$	$(i+1, j+1)$

Figure 2: Window size $(w \times w)$ with center pixel (i, j).

3.2 Feature Selection

Feature selection is a process of selecting relevant features which is used in classification model to perform the task. The pixel level color and texture features are computed from the transformed LAB color image. Section 3.2.1 and 3.2.2 explains the feature selection process for color and texture respectively.

3.2.1 Pixel Color Feature

Color is the most distinguishable feature in the image. The pixel level color feature [31] is computed using local homogeneity. Let (i, j) be the pixel in the input image of size $(M \times N)$ and T_{ij} be the transformed LAB color value for the pixel (i, j). T_{ij} has three color components such as $T_{ij}^{\mathrm{L}}, T_{ij}^{\mathrm{A}}$ and T_{ij}^{B}. Many homogeneous regions are formed as a result of segmentation. An image (I) is partitioned or divided into many subregions $(R_1, R_2, \ldots, R_n)$ based on predefined criteria. The sub regions should have uniformity and homogeneity with respect to the properties such as intensity, color, texture, other statistical properties, etc. The pixel level color features are calculated based on local homogeneity property of the image. The local information throughout the entire image is extracted by sliding window of size $(w \times w)$ with (i, j) as center pixel as shown in Fig. 2. The local homogeneity is calculated based on standard derivation of T_{ij} components using eq. (1).

The standard derivation (sd) for the center pixel (i, j) of the window $(w \times w)$ is computed for entire image $(M \times N)$ using eq. (1) [11]. Pixel color feature can be obtained by normalizing standard derivation values by using eq. (3).

$$sd_{ij}^{k} = \sqrt{\frac{1}{w^2} \sum_{m=i-\left(\frac{w-1}{2}\right)}^{i+\left(\frac{w-1}{2}\right)} \sum_{n=j-\left(\frac{w-1}{2}\right)}^{j+\left(\frac{w-1}{2}\right)} (T_{mn}^{k} - \mu_{ij}^{k})^2} \tag{1}$$

Figure 3: Image Sample.

where $1 \leq i,\ m \leq M,\ 1 \leq j,\ n \leq N$,
T_{mn}^{k} is LAB color components,
μ_{ij}^{k} is mean within the window $(w \times w)$ and calculated using eq. (2) from [11].

$$\mu_{ij}^{k} = \frac{1}{w^2} \sum_{m=i-\left(\frac{w-1}{2}\right)}^{i+\left(\frac{w-1}{2}\right)} \sum_{n=j-\left(\frac{w-1}{2}\right)}^{j+\left(\frac{w-1}{2}\right)} T_{mn}^{k} \tag{2}$$

$$sd_{ij}^{k} = \frac{sd_{ij}^{k}}{\max\{sd_{ij}^{k}\}} \tag{3}$$

3.2.2 Texture Feature

Texture and color features provide better segmentation [13, 31]. Textures are made up of texon which produces a pattern across it and shows uniformity among them. Texture features are estimated for the color components $T_{ij}^{k}(k = \mathrm{L, A, B})$ using range filter. Range filter is used to extract the local range from the specified window $(w \times w)$. Range filter is done using eq. (4), by finding out the difference between the maximum and minimum values in the specified neighborhood. The computed range filter values are normalized using eq. (5). The range filter is applied to the image shown in Fig. 3 and the resultant texture is shown in the Fig. 4.

$$rf_{ij}^{k} = \sum_{m=i-\left(\frac{w-1}{2}\right)}^{i+\left(\frac{w-1}{2}\right)} \sum_{n=j-\left(\frac{w-1}{2}\right)}^{j+\left(\frac{w-1}{2}\right)} \max_{mn}^{k} - \min_{mn}^{k} \tag{4}$$

$$rf_{ij}^{k} = \frac{rf_{ij}^{k}}{\max\{rf_{ij}^{k}\}} \tag{5}$$

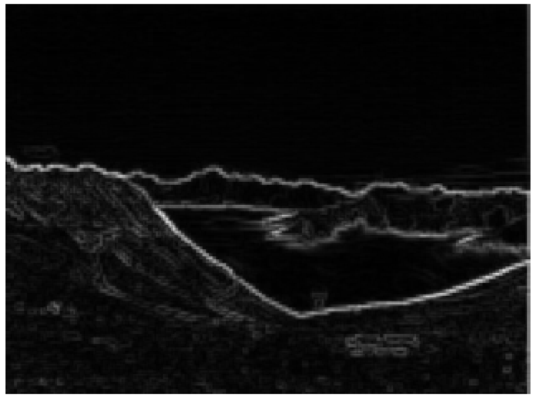


Figure 4: Texture Feature using Range filter.

3.2.3 Feature Vector

Feature vector is an n-dimensional vector which has numerical features of the image. The pixel color and texture features are computed from eq. (3) and eq. (5) respectively. The feature vector for each pixel in the image is constructed using local homogeneity and range filter using eq. (6) from [11, 26].

$$fv_{ij}^{k} = \begin{pmatrix} sd_{ij}^{k} \\ rf_{ij}^{k} \end{pmatrix} \tag{6}$$

3.3 Fuzzy C-Means (FCM) Clustering

Clustering is an unsupervised learning algorithm which is applied for classification problems. FCM algorithm has been widely adopted for image segmentation problems [24, 31]. The FCM algorithm performs well for noiseless images. In FCM, number of clusters (c_n) and cluster center for each cluster are initialized. Let X be set of data points ($X = \{x_1, x_2, \ldots, x_n\}$). Membership value is assigned to all data points. The center of cluster and membership values are updated by minimizing the objective function using eq. (7) from [30].

$$fcm(X,U,C) = \sum_{i=1}^{c_n} \sum_{j=1}^{n} u_{ij}^{m} (\|x_j - v_i\|)^2 \tag{7}$$

where v_i is the center of ith cluster,
u_{ij} is the membership value of the x_j data belonging to ith cluster,
$\|\|$ denotes Euclidean distance to measure similarity of data,
m is weighted exponent on each fuzzy membership,
U is max (membership value of all clusters) of x_j data and
C is centeriod of clusters.

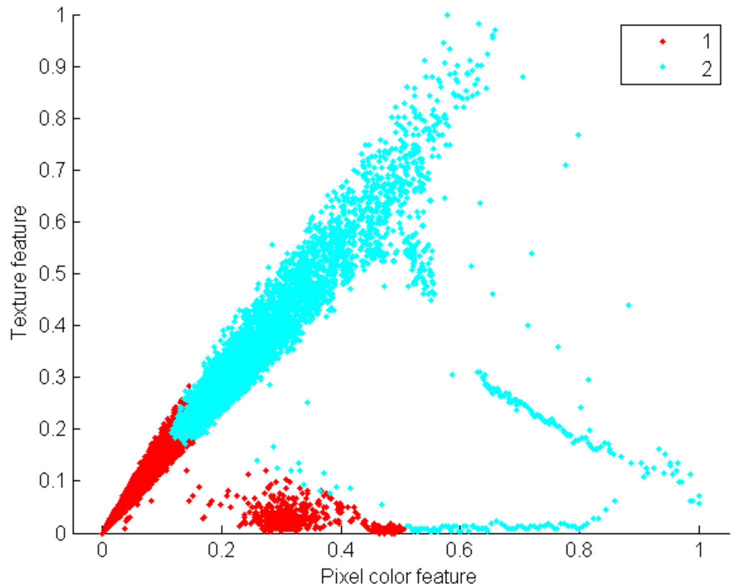


Figure 5: Clusters distribution for image shown in Fig. 3.

3.3.1 Image Segmentation using FCM

FCM is applied to cluster the pixels based on color and texture features. Membership value (u_{ij} where $i = 1$ to c_n (number of clusters) and $j = 1$ to n (number of pixels)) can be obtained for each data item with respect to each cluster. The pixels are classified based upon maximum membership value of all clusters using eq. (8).

$$u_{ij} = \max\{u_{1j}, u_{2j}, u_{3j}, \ldots, u_{cnj}\} \tag{8}$$

where jth pixel is assigned to ith cluster.

For example, consider the image shown in Fig. 3. FCM is applied to this image with 2 clusters. The pixels are classified into two clusters based on the feature vectors (color and texture). The clusters are shown in Fig. 5.The initial cluster centeroids are updated by minimizing the objective function. For the sample image in Fig. 3, final cluster centeroids are formed after 10th iteration which could be inferred from the graph Fig. 6. The result of FCM is segmented image which is shown in Fig. 7.

3.4 Pixel classification using Supervised Learning Methods

The segmentation is a classification technique where several groups are formed for every pixel from the classifier result. In case of supervised learning, class label is needed for classification. This method uses two phases namely

1. Training Phase—To train the classifier algorithm with the data matrix and label.
2. Testing Phase—To classify the new data and predict the class label for the test data.

The image can be segmented based upon their pixel level. The following supervised learning methods are applied for image segmentation problem to attain more accuracy and to reduce misclassification error.

1. Classification tree,

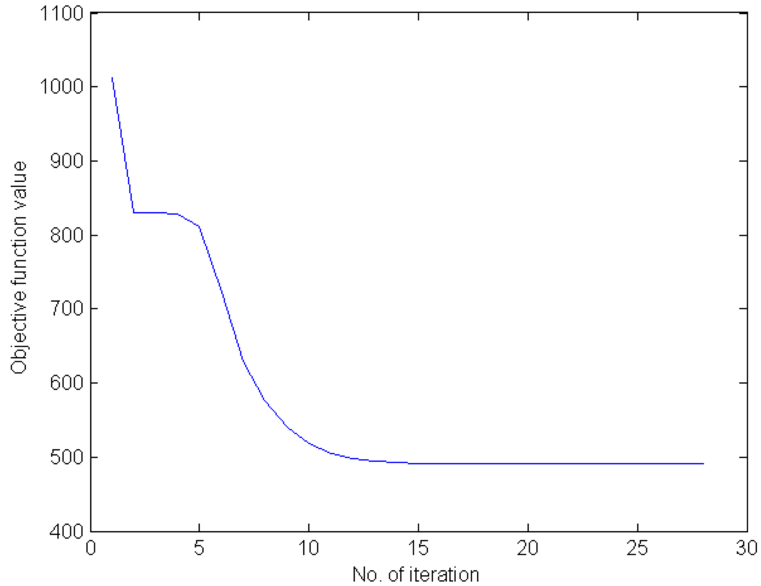


Figure 6: Convergence of FCM.

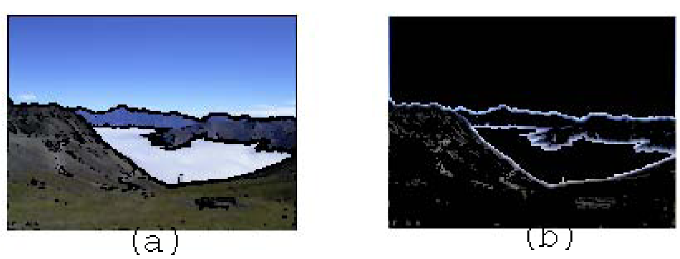


Figure 7: (a) Color Image segmentation result. (b) Gray Image segmentation result. Segmented image using FCM with 2 clusters.

2. SVM and
3. Adaboost

3.4.1 Construction of Training Samples for Image Segmentation

Training sample contains data matrix (feature vectors with n number of observations) and a class label vector for each observation (indicates vector belonging to the corresponding features). This paper uses color and texture features as discussed in Section 3.2 and number of observation is equal to number of pixels. The class label has to be assigned to each pixel by applying FCM as explained in Section 3.3. The data are selected based upon equal sampling rate. Out of n data (number of pixels in the image), 10 percentage of data from each cluster is selected using eq. (9) which is treated as training sample and the remaining data are kept for testing purpose.

$$training = \frac{ndata_i}{10} \tag{9}$$

where i denotes clusters from 1 to cn;
$ndata_i$ denotes number of data in ith cluster.

3.4.2 Classification Tree

This method constructs a binary tree for a given data matrix, based on the input features. Two phases of the classification tree are explained in Sections 3.4.2.1 and 3.4.2.2 respectively.

3.4.2.1 Training The classification tree method constructs a tree for the ensemble based on splitting criteria. The splitting criteria use Gini Diversity Index (GDI) of a node as given in eq. (10) from [33]. The error and purity of the node can be measured by their GDI value.

$$1 - \sum_i p^2(i) \tag{10}$$

where $p(i)$ is the observed fraction of classes for the node.

3.4.2.2 Testing The trained samples are tested in the testing phase. The predictor either reaches any one of the leaf nodes or misclassification occurs. When the predictor reaches a leaf node, it returns the corresponding node as class label. Else misclassification occurs. The predictor returns the label, where maximum number of training samples reaches that particular node.

3.4.2.3 Image segmentation by Classification Tree The training sample was constructed for Fig. 3 as explained in Section 3.4.1 with cluster size of 2. Out of 74730 data (number of pixels in the image), 7471 data are used for training by using eq. (9). Binary tree was constructed for the ensemble as shown in Fig. 8 with pruning levels of 6. The root node shows the total number of data used for training and also represents number of data belonging to clusters 1 and 2. Out of 7471 training data, 7071 data are classified as group1 data and the remaining 424 data items are classified as group2.

The error mainly depends on number of nodes (leaf) used for classification. The predictor classifies the data based on the value present in the leaf node as explained in the Section 3.4.2.2. If number of leaf nodes is more in the constructed tree, then error will be more as inferred from Fig. 9. Hence select the best pruning level to reduce the number of leaf nodes. The error difference between best pruning and non-pruning level is very less. The best pruning level is calculated and it can be used for classification. The best level is 4 and the tree is constructed for it as shown in Fig. 10. The number of leaf nodes is reduced to 7 with pruning level 4 whereas in Fig. 8 it has 30 leaf nodes for pruning level 6.

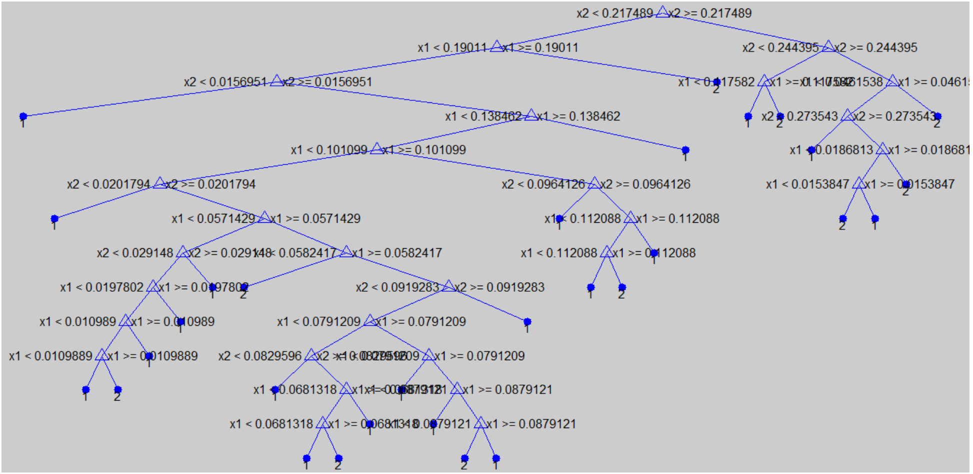


Figure 8: Classification tree for the ensemble shown in Fig. 3.

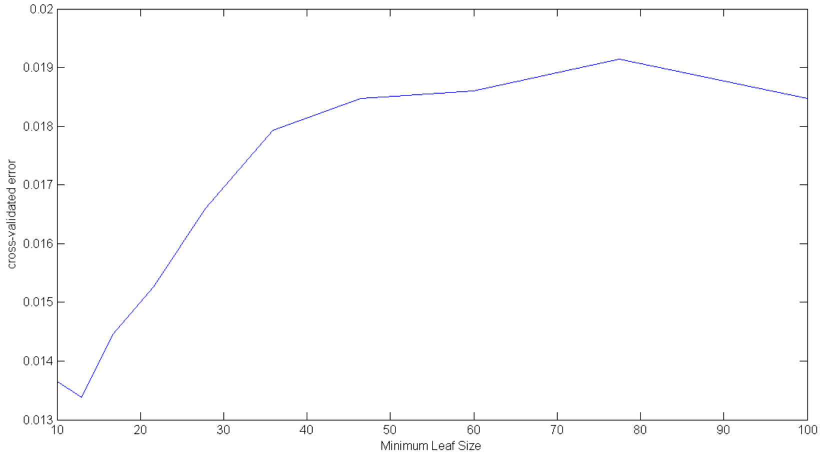


Figure 9: Number of leaf node Vs Error rate.

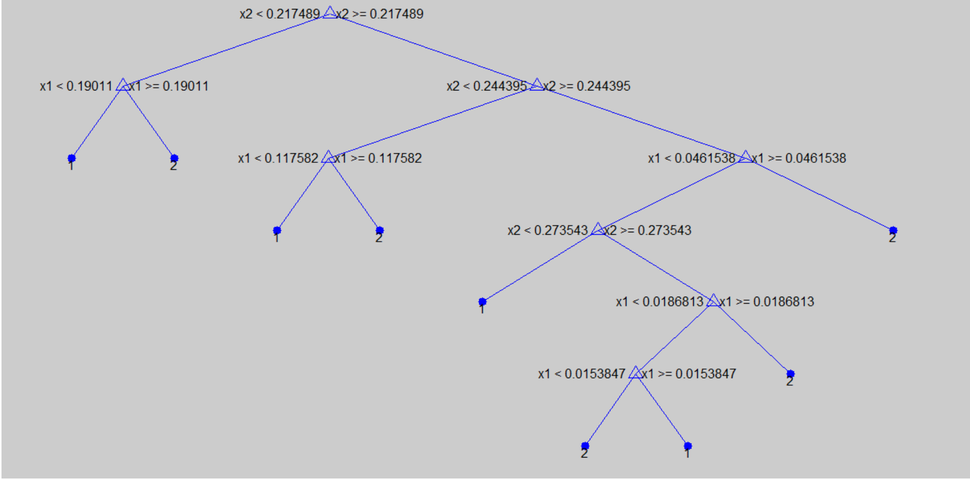


Figure 10: Classification tree for the ensemble shown in Fig.3 with best pruning level.

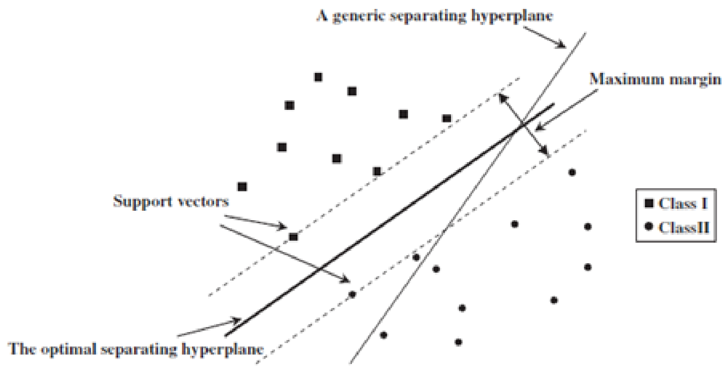


Figure 11: Linearly separable hyper plane for binary classification problem.

3.5 Support Vector Machine (SVM)

The aim of SVM is to find out the maximum hyperplane. The data may be linearly separable, linearly inseparable or non linearly classifiable. Almost all the image samples considered in this paper create clusters which are linearly separable as inferred from Fig. 5. An example of linearly separable classifier with two labels [26] is shown in Fig. 11. The following Sections 3.5.1 and 3.5.2 discuss the training and testing phase of linearly separable classification.

3.5.1 Training

The dual form equations of classifier [26] use eq. (11) to eq. (14) for linearly separable data. The hyper plane is found out by maximizing the objective function using eq. (11). The objective function is a convex constrained optimization problem. Eq. (12) gives the constraints for the objective function. The vectors which are present in the hyperplane are called as support vectors. After training, support vector (α_i) is find out which has the value greater than or equal to 0. By making use of the values of support vector, bias (b) value can be found out by substituting the values in eq. (13).

$$\max L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum y_i y_j \alpha_i \alpha_j x_i^T x_j \tag{11}$$

$$\sum \alpha_i y_i = 0 \ \& \ 0 \leq \alpha_i \tag{12}$$

$$b = \frac{1}{Nsv} \sum_{sv \in S} \left(y_{sv} - \sum_{i \in S} \alpha_i y_i x_i x_{sv} \right) \tag{13}$$

where x_{sv} is index of the support vector, x_i is the data, y_i is class level
b is bias, α_i is suport vector, Nsv indicates the number of support vectors.

3.5.2 Testing

The new data is tested using the classifier equation $f(y)$ as mentioned in eq. (14). The sign of the classifier equation decides the class label for the predictor. For eq. (15), it is inferred that if $f(y)$ has

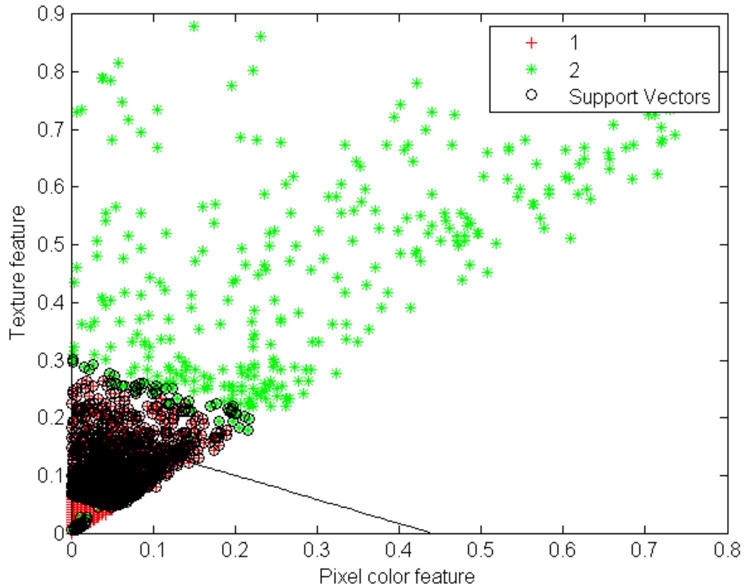


Figure 12: Training image sample of Fig. 3 on SVM.

negative value then the data belongs to the class label 1. Else the data belongs to 0 class.

$$f(y) = \text{sign}\left(\sum_{i=1}^{sv} \alpha_i y_i x_i x_{sv} + b\right) \tag{14}$$

$$p_i = \begin{cases} 1 & \text{iff } (y_i) \text{ is negative} \\ 0 & \text{iff } (y_i) \text{ is positive} \end{cases} \tag{15}$$

where p_i is the predicted class label for the predictor.

3.5.3 Image Segmentation using SVM

The sample data constructed from Section 3.4.1, is used as training data for the input image shown in Fig. 3. The SVM model is trained with the data sample and class label, it gives bias and support vectors value. Fig. 12 shows the data sample, hyperplane and their support vectors. It classifies the test data by making use of the results obtained in training phase. All the test data are labeled by using eq. (15). The result of the tested sample is shown in Fig. 13.

3.6 Adaboost Algorithm

The adaboost algorithm uses bunch of weak classifier to provide stronger classification result. The algorithm uses minimization of exponential error. The framework of Adaboost algorithm is shown in Fig. 14. The training data, labels and weaker classifier (decision tree) are given to fit ensemble to get the ensembles for data. The training and testing of adaboost algorithm is explained in Sections 3.6.1 and 3.6.2 respectively.

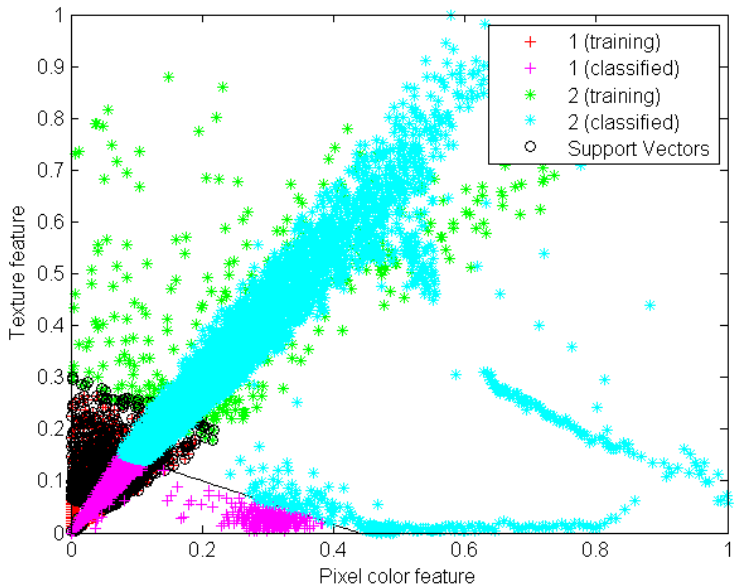


Figure 13: Testing image (Fig. 3) sample on SVM.

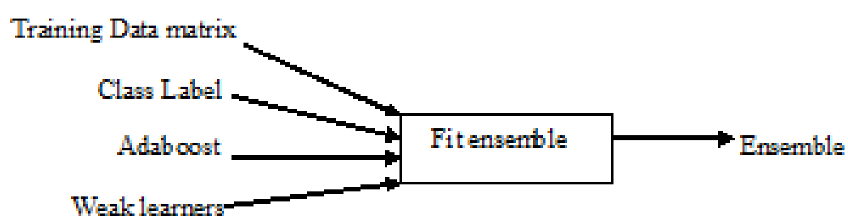


Figure 14: Frame work of Adaboost Algorithm.

3.6.1 Training

Adaboost algorithm [27,32] trains n number of weak learner with t index. The weights $w_{1,i}$ are initialized using eq. (16). The weaker classifiers h_j for each feature are trained and h_j is chosen such that it should contain minimum error for the eq. (17) from [32]. The weights are updated using eq. (18). Finally stronger classifier is obtained from bunch of weaker classifiers.

$$w_{1,i} = \begin{cases} \frac{1}{2m} & \text{if } i \text{ is positive sample} \\ \frac{1}{2l} & \text{if } i \text{ is negative sample} \end{cases} \tag{16}$$

where m is positive sample and l is negative sample

$$e_i = \sum_i w_{t,i} |h_i(x_i) - y_i| \tag{17}$$

where e_i is the error, x_i is the sample data, y_i is class label.

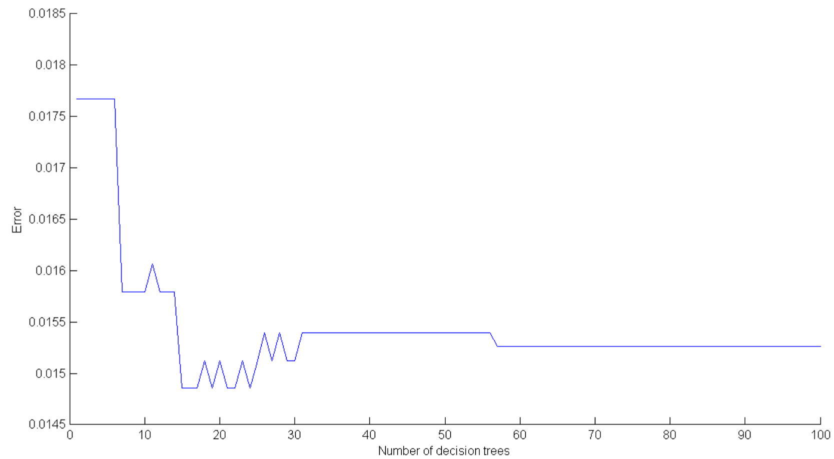


Figure 15: Error Vs Number of decision tree.

The weights are updated using eq. (18) with calculated error e_i using eq. (19) referred from [32]

$$w_{t+1,i} = w_{t,i}\beta_t^{1-e_i} \tag{18}$$

$$e_i = \begin{cases} 1 & \text{if } x_i \text{ classified correct} \\ 0 & \text{if } x_i \text{ classified incorrect} \end{cases} \tag{19}$$

3.6.2 Testing

The class for new data (testing data) can be predicted based upon the eq. (20) referred from [32]. The data can be predicted as class 1 or 0 for H(x).

$$H(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(x) > \frac{1}{2}\sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

where $\alpha_t = \log \frac{1}{\beta_t}$ which is the weight of the weaker learner in the ensemble.

3.6.3 Image segmentation using Adaboost

The ensemble samples of Fig. 3 are trained using Adaboost algorithm as discussed in Section 3.6. It uses tree as a weaker classifier and binary classifier (Labels are either 1 or 2). It is inferred from the Fig. 15 that the error rate varies with respect to number of trees (weak learner) used. Number of weaker learners should provide less error which means that the classification accuracy is more. The constant error is attained with 60 trees for the data set obtained for input image of Fig. 3.

4 Results and Performance Analysis

The methods discussed in the Section 3 were implemented using MATLAB 12 in Intel i7 processor @ 2.67GHz. The data used for the experiment is collected from Berkeley segmentation dataset [1]. It has almost 300 benchmark images. All the images of the database are trained and tested based on the three different algorithms such as Classification tree, SVM and Adaboost algorithms. Some of the benchmark

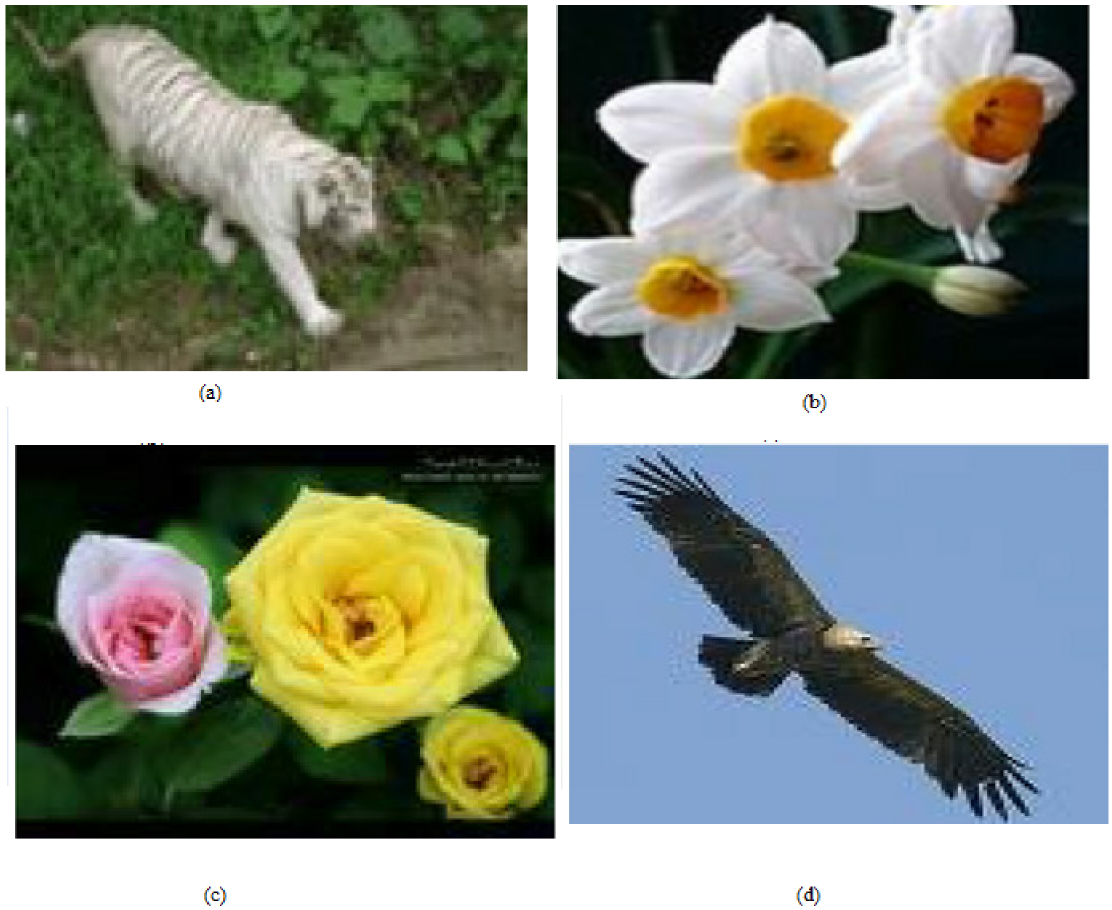


Figure 16: Sample Images used for classification.

images are shown in Fig. 16. The performances of the methods are compared based on their error rate (misclassification), time taken for training and testing the sample.

4.1 Confusion Matrix

A matrix, called confusion matrix, is constructed based on the class labels of actual (observed) and predicted (tested) result as shown in Table 1. The classification and misclassification of data are found out from the confusion matrix, where TN and FP values indicate misclassification of data and TP and FN indicate correct classification of data. The error and accuracy [11] can be calculated using eq. (21) and eq. (22) respectively.

$$ce = \frac{(FP + TN)}{N} \tag{21}$$

$$ca = \frac{(TP + FN)}{N} \tag{22}$$

where TP is True Positive, TN is true Negative, FP is False Positive and FN is False Negative, N is total number of pixels in the image,

$$ce = \text{Classification error} \quad \text{and} \quad ca = \text{Classification accuracy}$$

Table 1: Confusion matrix.

		Actual class (observed)	
		$+1$	-1
Predicted class (expected), where TP is	$+1$	TP	TN
	-1	FP	FN

Table 2: Confusion matrix for the Fig. 3.

Methods	Classification tree actual class		SVM actual class		Adaboost actual class	
Confusion matrix	69 163	1320	67 286	0	70 345	138
(predicted class)	289	3958	3 197	4247	595	3652

Table 2 shows the confusion matrix for Fig. 3. The matrix is created for three different methods such as Classification tree, SVM and Adaboost. From the result, it is observed that misclassification of data is much lower in Adaboost when compared to Classification Tree and SVM methods. The adaboost produces stronger classification result by making use of many weaker classifiers as discussed in 3.6.

4.2 Results of Segmentation

The segmentation code implemented for three classification methods are run in MATLAB by initializing the following parameters as constants:

1. Minimum local window size $(w \times w)$ is considered (i.e.) 3×3.
2. Number of clusters (c_n) is 2 because binary classifier is used.

The classification algorithms are tested for the benchmark images [1]. The segmented results for three methods (Classification tree, SVM, Adaboost) are shown in Fig. 17.

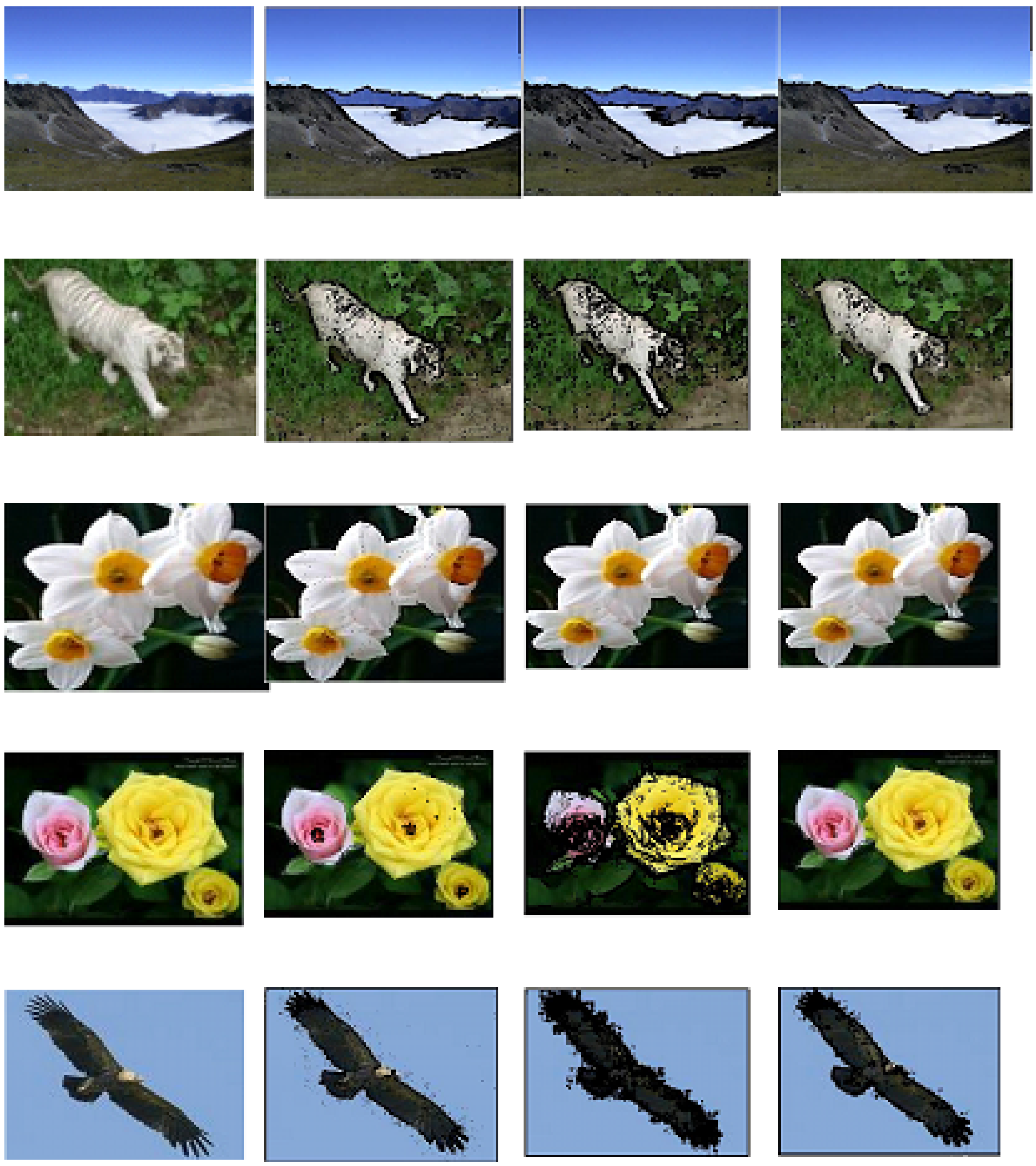
4.3 Performance Evaluation

The error was computed for all three classification methods using eq. (21), for the images shown in Fig. 16. The computational time was also calculated for two phases namely training and testing. The observed data were tabulated in Table 3. It is inferred from the Table 3 that the Adaboost has less error but takes more computation time for training and testing due to usage of many weak learners. The learning time increases exponentially with respect to the number of weaker classifiers.

From the implementation results shown in Fig. 18, it is inferred that Adaboost misclassification is less for all the image samples when compared to the Classification Tree and SVM methods. It is observed from Fig. 19 and Fig. 20, the training and testing time of Adaboost algorithm is more for all image samples when compared to other two supervised classification methods. This is due to many weaker classifiers required to acquire stronger classifier.

5 Conclusion

This paper proposed, implemented and tested a two level hybrid non classical model for image segmentation. The hybrid models were designed using FCM and supervised classification methods. The designed models were implemented to segment the image and their results were compared. From the experimental results, it is inferred that Adaboost correctly predicts most of the data samples, reduces error and increases accuracy when compared to Classification Tree and SVM.



a) Original Image b) Classification tree c) SVM d) Adaboost

Figure 17: (a) Original Image. (b) Classification tree. (c) SVM. (d) Adaboost. Segmented image based on Classification Tree, SVM and Adaboost.

Table 3: Comparison of classification tree, SVM, Adaboost.

Methods	Image 1 (Fig. 3)			Image 2 (Fig. 16(a))		
Performance mertics	Error (%)	Training time (s)	Testing time (s)	Error (%)	Training time (s)	Testing time (s)
Classification tree	2.15	0.66	0.9875	5.1	0.5524	0.10832
SVM	4.28	6.108	1.4319	9.7	10.4087	4.27803
Adaboost	1.1	10.1396	5.994	4.25	9.2928	5.52108
	Image 3 (Fig. 16(b))			Image 4 (Fig. 16(c))		
	1.71	0.292	0.1078	4.82	0.279091	0.0811
	1.48	3.070	0.6417	28.0	5.4735	2.55468
	1.0	9.6013	5.568	3.97 7.6349	5.5317	

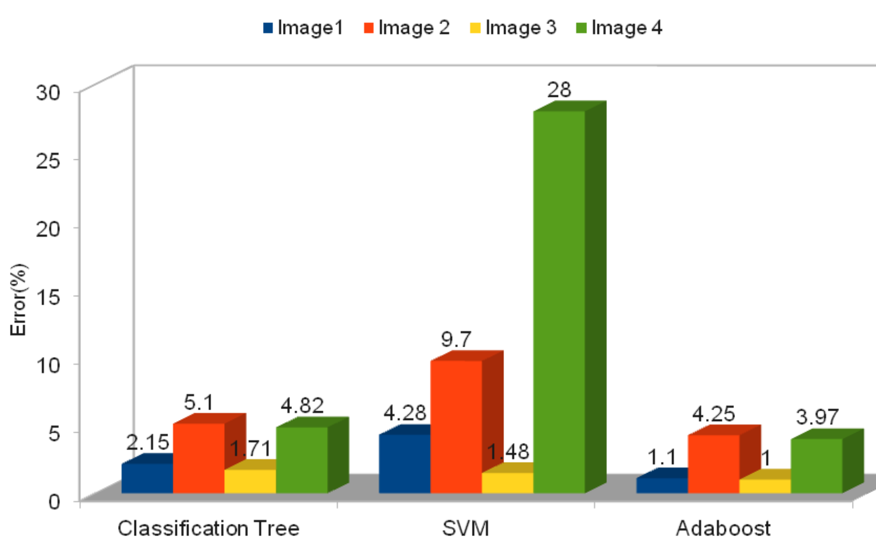


Figure 18: Error in Classification Tree, SVM and Adaboost for different images.

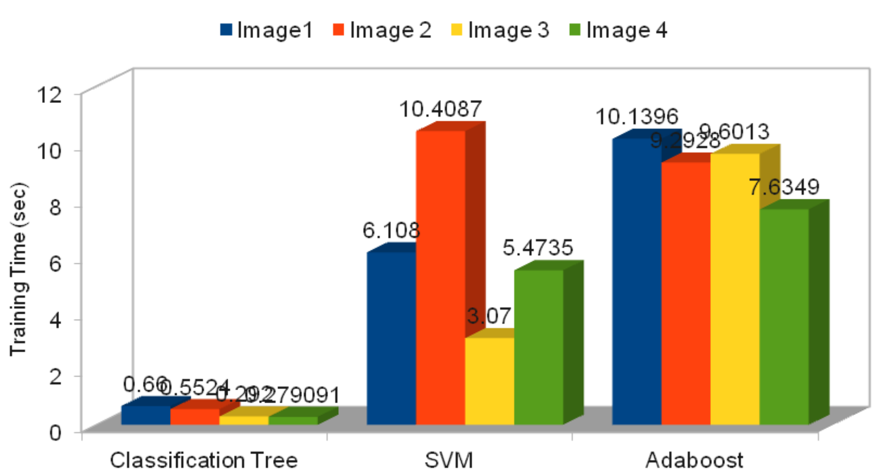


Figure 19: Training time for Classification Tree, SVM and Adaboost with different images.

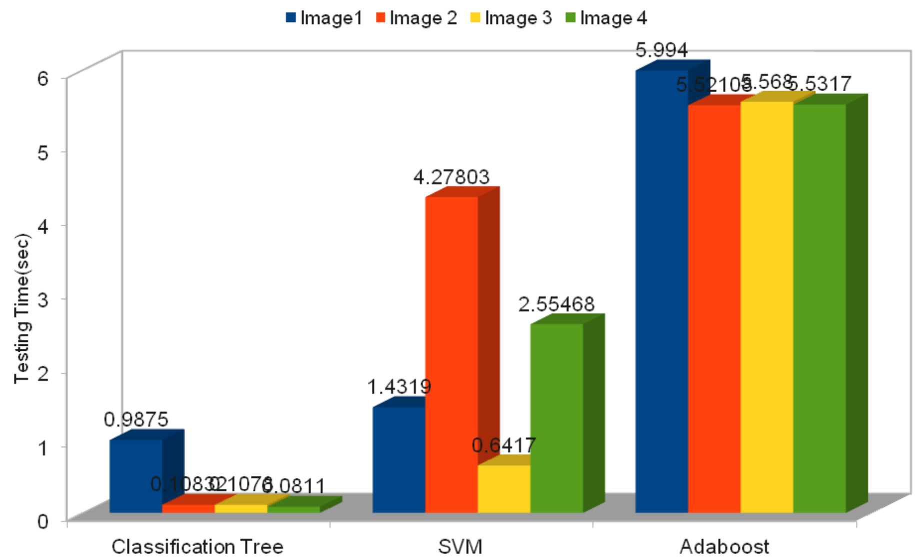


Figure 20: Testing time for Classification Tree, SVM and Adaboost with different images.

References

[1] P. Arbelaez, C. Fowlkes, and D. Martin. The berkeley segmentation dataset and benchmark, June 2007. `http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/`.

[2] X. Baia and W. Wang. Saliency-svm: An automatic approach for image segmentation. *Journal of Neurocomputing*, 136:243–255, July 2014.

[3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, November 2001.

[4] R.-F. Chang, W.-J. Wu, W. K. Moon, and D.-R. Chen. Improvement in breast tumor discrimination by support vector machines and speckle-emphasis texture analysis. *Journal of Image and Vision Computing*, 29(5):679–686, May 2003.

[5] G. Chen and W. Xie. Pattern recognition with svm and dual-tree complex wavelets. *Journal of Image and Vision Computing*, 25(6):960–966, June 2007.

[6] C. M. Christoudias, B. Georgescu, and P. Meer. Synergism in low-level vision. In *Proc. of the 16th International Conference on Pattern Recognition (ICPR'02), Québec, Canada*, pages 150–155. IEEE, August 2002.

[7] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24(5):603–619, May 2002.

[8] B. Cyganek. Color image segmentation with support vector machines: applications to road signs detection. *Journal of Neural Systems*, 18(4):339–345, August 2008.

[9] J.-B. Fiot, L. D. Cohen, P. Raniga, and J. Fripp. Efficient brain lesion segmentation using multi-modality tissue-based feature selection and support vector machines. *Journal of Numerical Methods in Biomedical Engineering*, 29(9):905–915, September 2013.

[10] H. Fleyeh, R. Biswas, and E. Davami. Traffic sign detection based on adaboost color segmentation and svm classification. In *Proc. of the IEEE EuroCon (EUROCON'13), Zagreb, Croatia*, pages 2005–2010. IEEE, July 2013.

[11] R. C. Gonzalez and S. L. E. Richard E. Woods. *Digital Image Processing*. Pearson Education, 2010.

[12] D. Gupta, A. K. Malviya, and S. Singh. Performance analysis of classification tree learning algorithms. *Journal of Computer Applications*, 55(6):39–44, October 2012.

[13] D. E. Ilea and P. F. Whelan. Image segmentation based on the integration of color-texture descriptors—a review. *Journal of Pattern Recognition*, 44(10-11):2479–2501, October–November 2011.

[14] A. K. Jain and F. Farrokhnia. Unsupervised texture segmentation using gabor filters. In *Proc. of the IEEE International Conference on System, Man and Cybernetics (ICSMC'90), Los Angeles, USA*, pages 14–19. IEEE, November 1990.

[15] C.-F. Juang, S.-H. Chiu, and S.-J. Shiu. Fuzzy system learned through fuzzy clustering and support vector machine for human skin color segmentation. *IEEE Transaction on Systems, Man and Cybernetics*, 37(6):1077–1087, November 2007.

[16] M. Krishna and A. Srinivasulu. Face detection system on adaboost algorithm using haar classifiers. *Journal of Modern Engineering Research*, 2(5):3556–3560, September-October 2012.

[17] F. Malmberga, J. Lindbladb, N. Sladojec, and I. Nyströma. A graph-based framework for sub-pixel image segmentation. *Journal of Theoretical Computer Science*, 412(15):1338–1349, February 2011.

[18] J. Ninga, L. Zhanga, D. Zhanga, and C. Wu. Interactive image segmentation by maximal similarity based region merging. *Journal of Pattern Recognition*, 43(2):445–456, February 2010.

[19] J. Ninga, L. Zhanga, D. Zhanga, and C. Wu. Multivariate image segmentation using semantic region growing with adaptive edge penalty. *Journal of Pattern Recognition*, 19(8):2157–2170, August 2010.

[20] J.-J. Quan, X.-B. Wen, and X.-Q. Xu. Multiscale probabilistic neural network method for sar image segmentation. *Journal of Applied Mathematics and Computation*, 205(2):578–583, November 2008.

[21] M. B. Salah, A. Mitiche, and I. B. Ayed. Multiregion image segmentation by parametric kernel graph cuts. *IEEE Transaction on Image Processing*, 20(2):545–557, February 2011.

[22] B. Sowmya and B. S. Rani. Colour image segmentation using fuzzy clustering techniques and competitive neural network. *Journal of Applied Soft Computing*, 11(3):3170–3178, April 2011.

[23] M. Sridevi and C. Mala. A survey on monochrome image segmentation methods. In *Proc. of the 2nd International Conference on Communication, Computing Security (ICCCS'12), Rourkela, India*, pages 548–555. Elsevier, 2012.

[24] K. S. Tan and N. A. M. Isa. Color image segmentation using histogram thresholding—fuzzy C-means hybrid approach. *Journal of Pattern Recognition*, 44(1):1–15, January 2011.

[25] L. G. Ugarriza, E. Saber, S. R. Vantaram, V. Amuso, M. Shaw, and R. Bhaskar. Automatic image segmentation by dynamic region growth and multiresolution merging. *Journal of IEEE Transaction on Image Processing*, 18(10):2275–2288, October 2009.

[26] V. Vapnik. *The nature of statistical learning theory*. Springer, 1999.

[27] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01), Hawaii, USA*, pages I–511–I–518. IEEE, November 2001.

[28] J. Wang, L. Ju, and X. Wang. An edge-weighted centroidal voronoi tessellation model for image segmentation. *Journal of IEEE Transaction on Image Processing*, 18(8):1844–1858, August 2009.

[29] H. Wanga and J. Oliensis. Generalizing edge detection to contour detection for image segmentation. *Journal of Computer Vision and Image Understanding*, 114(7):731–744, July 2010.

[30] X.-Y. Wanga and J. Bu. A fast and robust image segmentation using fcm with spatial information. *Journal of Digital Signal Processing*, 20(4):1173–1182, July 2010.

[31] X.-Y. Wanga, T. Wanga, and J. Bu. Color image segmentation using pixel wise support vector machine classification. *Journal of Pattern Recognition*, 44(4):777–787, April 2011.

[32] t. f. e. Wikipedia. Adaboost(wikipedia), August 2014. `http://en.wikipedia.org/wiki/AdaBoost`.

[33] t. f. e. Wikipedia. Decision tree learning(wikipedia), July 2014. `http://en.wikipedia.org/wiki/Decision_tree_learning`.

[34] Z. Yua, H.-S. Wongb, and G. Wena. A modified support vector machine and its application to image segmentation. *Journal of Image and Vision Computing*, 29(1):29–40, January 2011.

[35] B. Zheng, S. W. Yoon, and S. S. Lam. Breast cancer diagnosis based on feature extraction using a hybrid of K-means and support vector machine algorithms. *Journal of Expert Systems with Applications*, 41(4):1476–1482, March 2014.

Author Biography

C. Mala is currently serving as an Associate Professor in the Department of Computer Science and Engineering, National Institute of Technology, Trichy, India. She received Ph.D. from National Institute of Technology, Trichy in 2008. Her research interests include Wireless Networking, Parallel Algorithms, Soft Computing and Image Processing.

M. Sridevi received M.E. degree in Computer Science and Engineering from Annamalai University, India in 2006. She is working as an Assistant Professor in National Institute of Technology (NIT) Trichy since 2008 and currently pursuing Ph.D. degree at NIT, Trichy. Her research interests include Image Processing, Parallel Algorithms and Optimization Techniques.