

Colour Image Segmentation using Fast Fuzzy C-Means Algorithm

Kishor Bhoyar* and Omprakash Kakde⁺

** Department of Information Technology, YCCE, Nagpur, India*

+ Department of Computer Science and Engineering, VNIT Nagpur, India

Received 11th July 2009; accepted 5th July 2010

Abstract

This paper proposes modified FCM (Fuzzy C-Means) approach to colour image segmentation using JND (Just Noticeable Difference) histogram. Histogram of the given colour image is computed using JND colour model. This samples the colour space so that just enough number of histogram bins are obtained without compromising the visual image content. The number of histogram bins are further reduced using agglomeration. This agglomerated histogram yields the estimation of number of clusters, cluster seeds and the initial fuzzy partition for FCM algorithm. This is a novell approach to estimate the input parameters for FCM algorithm. The proposed fast FCM(FFCM) algorithm works on histogram bins as data elements instead of individual pixels. This significantly reduces the time complexity of FCM algorithm. To verify the effectiveness of the proposed image segmentation approach, its performance is evaluated on Berkeley Segmentation Database(BSD). Two significant criteria namely PSNR(Peak Signal to Noise Ratio) and PRI (Probabilistic Rand Index) are used to evaluate the performance. Although results show that the proposed algorithm applied to the JND histogram bins converges much faster and also gives better results than conventional FCM algorithm, in terms of PSNR and PRI.

Key Words: Colour Image Segmentation, JND Histogram, Fuzzy C-means Clustering, Fast FCM

1 Introduction

Segmentation involves partitioning an image into a set of homogeneous and meaningful regions, such that the pixels in each partitioned region posses an identical set of properties. Fuzzy set theory has been extensively applied in the area of image segmentation. The concepts of fuzzy first order and second order statistic like fuzzy histogram and co-occurrence matrices have been presented in [1,2]. These measures have been demonstrated to yield excellent segmentation results for the images having bimodal or multimodal histograms. The fuzzy second order statistics characterize textured images in a better way than their corresponding hard counterparts [3]. A number of research reports indicate the superiority of fuzzy sets in segmenting an image over its crisp counterparts.

Fuzzy approaches to pixel classification have found applications in problems where precise knowledge about the pattern classes is not available, large number of pattern samples are not available for statistical estimation of parameters and patterns have partial membership to different classes. In many real world images, the clusters are not disjoint and a simple pattern or a pixel in an image may belong to different clusters. For example, a particular pixel in the sky may belong partly to sky and partly to the cloudy-sky class. Fuzzy approaches to supervised pattern classification and clustering may be found in [4]. Clustering methods are considered as an unsupervised classification technique, where there is no need for prior

Correspondence to: <kkbhoyar@yahoo.com>

Recommended for acceptance by <Jean Ogier>

ELCVIA ISSN: 1577-5097

Published by Computer Vision Centre / Universitat Autònoma de Barcelona, Barcelona, Spain

knowledge about the data set. Many different clustering techniques in computer vision and pattern recognition have been proposed in [5,6]. Among those techniques FCM algorithm is widely used for image segmentation with some modifications. FCM algorithm allows points to belong to two or more classes with appropriate membership. This flexibility during the classification process is a useful property since it informs about the confidence attributed to a particular class or cluster. Although the original FCM algorithm yields good segmenting results on noise-free images, it fails to segment images corrupted by noise, and other imaging artefacts [7]. To overcome these problems a modified FCM segmentation approach based on spatial constraints is presented in [7, 8]. Attempt to reveal inherent non-Euclidean structures in image data has been discussed in [8]. In [9] a robust fuzzy clustering based segmentation method for noisy images is developed by introducing a novel penalty term into the objective function. In [25] the modified Fuzzy C-Means (FCM) algorithm is proposed to compensate for intensity non-homogeneities by modifying the objective function. Fast Generalized Fuzzy C-means clustering algorithm (FGFCM) for incorporating local spatial and gray information based on actual gray levels present in an image has been proposed in [24].

FCM clustering algorithm has been extensively used in various applications for image segmentation. Few recent references applied them on colour images of CT SCAN [25] and MRI [26], in the medical imaging successfully. The modified FCM algorithm [7] mentioned above has been successfully applied on remote sensing images as well as synthetic images. In [27], a remote sensing image segmentation procedure that utilizes a single point iterative weighted fuzzy C-Means clustering algorithm is proposed based on the prior information.

Our paper is unique in the sense that, we have applied FCM and its proposed variant FFCM on Berkeley Image Dataset of 300 colour images[20] and evaluated its performance in terms of two important performance measures for image segmentation, namely PRI and PSNR (defined in section 5). The classical FCM algorithm requires that the desired number of clusters be given in advance. This can be a problem when the clustering problem does not specify any desired number of clusters [25]. In this regard, we have proposed a novel scheme for initialization of FCM clustering which provides this information using colour similarity approach based on human perception discussed in section 2 and section 4.

The rest of the paper is organized as follows. Section 2 gives the brief overview of JND model in RGB space and computation of JND histogram. Section 3 presents the algorithm for agglomeration of JND histogram and its analysis that provides valuable information to FCM algorithm like the number of clusters and cluster seeds. Section 4 presents the FCM algorithm for colour image segmentation applied on individual pixels and its time efficient variant applied to the groups of similar pixels (bins) of agglomerated JND histogram. Section 5 presents the results on BSD. Section 6 presents the concluding remarks.

2 JND colour model and JND histogram

2.1 Overview of JND colour model

JND (Just Noticeable Difference) also known as JNCD (Just Noticeable Colour Difference) is defined as the smallest difference between two stimuli that a person can detect. The term JND is first coined by E. Goldstein and S. Fink [10] in their article, "Selective Attention in Vision", in the Journal of Experimental Psychology in 1981. It is the visibility threshold below which any change in colour cannot be detected by the Human Vision System (HVS)

The JND colour model in RGB space based on limitations of human vision perception as proposed in [11] is briefed here for ready reference. The human retina contains two types of light sensors, namely: rods and cones, responsible for monochrome i.e. gray vision and colour vision respectively. The three types of cones viz, Red, Green and Blue respond to specific ranges of wavelengths corresponding to the three basic colours Red, Green and Blue. The concentration of these colour receptors is maximum at the centre of the retina and it goes on reducing along radius. According to the three colour theory of Thomas Young [12], all other colours are perceived as linear combinations of these basic colours. According to [12] a normal human eye can perceive at the most 17,000 colours at maximum intensity without saturating the human eye. In other words, if the huge colour space is sampled in only 17,000 colours, a performance matching close to human vision at normal illumination may be obtained. A human eye can discriminate between two colours if they are at least one 'just noticeable difference' away from each other. The actual value of the just noticeable difference in terms of colour co-ordinates may not be constant over the complete RGB space due to non-

linearity of human vision and the non-uniformity of the RGB space. The CIELAB [3] colour space has been adopted as an international standard in the 1970s, by CIE (Commission Internationale de l'Éclairage) [3]. It is a perceptually uniform space, meaning that the Euclidean distance between two colour points in the space corresponds to the perceptual difference between the two colours, perceived by the human visual system. However, the term 'just noticeable difference (JND)' has been qualitatively used as a colour difference unit [13].

From the physiological knowledge, the red cones in the human retina are least sensitive, blue cones are moderately sensitive and the green cones are most sensitive. Keeping this physiological fact in mind, the red axis has been quantized in 24 levels and the blue and green axes are quantized in 26 and 28 levels [11]. The 24x26x28 quantization in the RGB space results in slight over-sampling (17,472 different colours) but it ensures that each of the 17,000 colours is accommodated in the sampled space. Heuristically it may be verified that any other combination of quantization on the R,G and B axes results in either large under sampling or over-sampling as required to accommodate 17000 colours in the space.

Using this sampling notion and the concept of 'just noticeable difference' the complete RGB space is mapped on to a new colour space $J_r J_g J_b$ where J_r , J_g and J_b are three orthogonal axes which represent the Just Noticeable Differences on the respective R,G and B axes. The values of J on each of the colour axes vary in the range (0,24) ,(0,26) or (0,28) respectively for red, blue and green colours. This new space is a perceptually uniform space and offers the advantages of the uniform spaces in image analysis.

2.2 Approximating the value of colour similarity threshold (JND)

For a perfectly uniform colour space the Euclidean distances between two colours is correlated with the perceptual colour difference. In such spaces (e.g. CIELAB to a considerable extent) the locus of colours which are not perceptually different from a given colour, forms a sphere with a radius equal to JND. As RGB space is not a perceptually uniform space, the colours that are indiscernible from the target colour, form a perceptually indistinguishable region with irregular shape. We have tried to derive approximate value of JND by 24 x 26 x 28 quantization of each of the R, G, and B axes respectively. Thus, such perceptually indistinguishable irregular regions are modelled by 3-D ellipsoids for practical purposes.

The research in physiology of human eye indicates two types of JND factors involved in the human vision system. The first is the JND of human eye referred to as JND_{eye} and the second is the JND of human perception referred to as JND_h . It is found that the neural network in human eye is more powerful and can distinguish more colours than those ultimately perceived by the human brain. The approximate relationship between these two [11] is given by equation 1:

$$JND_h = 3.JND_{eye} \quad \text{--- (1)}$$

Let C_1 and C_2 be two RGB colours in the new quantized space. Let $C_1 = (J_{r1}, J_{g1}, J_{b1}) = (0,0,0)$ and its immediate JND neighbour, that is 1 noticeable difference away is $C_2 = (J_{r2}, J_{g2}, J_{b2}) = (255/24, 255/28, 255/26)$. Hence $JND_{eye} = \sqrt{((255/24)^2 + (255/28)^2 + (255/26)^2)} = \sqrt{285.27}$. Using equation (1) the squared JND threshold of human perception is given by equation 2:

$$\Theta = JND_h^2 = 2567 \quad \text{--- (2)}$$

In equation (1), the squared distance is used, to avoid square root computation and hence to reduce time complexity. The use of θ as a squared threshold is very convenient mechanism to exploit perceptual redundancy inherent in digital images, as it gives opportunity to work in sampled colour space without compromising on the visual quality of results. For practical applications the range of θ for fine to broad vision is $JND_{eye}^2 \leq \theta \leq JND_h^2$. Similar calculations can be performed in HSV space.

2.3 Computing JND Histogram

Histogram of an image manifests its important global statistics, which can be used for a number of analysis and processing algorithms. In colour image histograms, a large number of colours may be present as

required for representing real life images. All of these colours may not even be noticed as different colours by normal human eye [11], hence as the first step the histogram on each of its axis has been sampled suitably to accommodate all the human distinguishable colours.

In this section, we propose an algorithm for computing histogram of a colour image in RGB space. As the structure of this histogram is four dimensional it is difficult to represent it in a 2-D plane and hence it is hardly possible to plot. Thus the histogram of an RGB image $I=f(1),f(2),f(3),\dots,f(m \times n)$ is given by $H(r,g,b)$ as in equation 3, where m and n are rows and columns of the image respectively and I represents the colour intensity values[14]. N is a counter variable and r, g, b represents the colour coefficients.

$$H(r, g, b) = \sum_{i=1}^{m \cdot n} N \Big|_{f=f(r,g,b)} \quad \text{--- (3)}$$

In the proposed histogram, the first data structure is a table of size $n \times 4$, where n indicates number of different colours in the image. Out of the four columns three are used for the RGB colour intensities and the fourth is for population of that colour. The second data structure is a table of $(r \times c)$ rows (where r and c indicate rows and columns in an image) and out of the three columns the first two are used for storing the colour position information of a pixel in terms of the x and y coordinates and the remaining one column is used for the colour index i.e. row number in the first data structure. In this form the colour image histogram becomes a solid cube while the density of the cube at a point in it represents the frequency and the three orthogonal edges of the cube represent the basic R, G and B colours. A traditional histogram does not contain any positional information. With the positional information stored in the proposed histogram, it just becomes transform of an image. In other words, the image can be obtained back from the new histogram with the positional information. The spatial colour distribution information also plays an important role in the image analysis. Both of these histogram tables have been shown in Table 1 and Table 2. These new histogram data structures will be collectively called as JND histogram.

For practical purposes already discussed, the colour image histograms have to be sampled on R, G and B axis suitably to reduce the number of colours. Most of the literature till now either uses uniform sampling of the R, G, B axis or uses images represented in uniform colour spaces. Such a uniformly sampled histogram can be represented by equation 4 with the same symbols. δ represents sampling interval on each axis and p is an integer variable.

$$H(p\delta r, p\delta g, p\delta b) = \sum_{i=1}^{m \cdot n} N \Big|_{f=f(p\delta r, p\delta g, p\delta b)} \quad \text{--- (4)}$$

The four dimensional colour image histogram is represented by two linked structures as given in Table 1 and Table 2.

JND Colour Index	R	G	B	H
1	5	15	20	335
2	10	100	20	450
3	20	50	10	470
.
k	25	72	90	200

Table 1: JND Histogram Structure

X	Y	JND Colour Index for (x_i, y_i) from Table 1
x_1	y_1	1
x_1	y_2	1
.	.	.
x_i	y_i	K
.	.	.

Table 2: Colour spatial structure

Thus Table 1 contains the R, G, B coordinates and the respective frequency information or population (H) of the tri-colour stimulus, while Table 2 contains the x and y positional co-ordinates in the image and respective colour index (row index) in Table 1. The number of rows in Table 1 is equal to the number of different colour shades available in the image. In Table 1 there will be one entry for each colour shade while in Table 2 there will be one entry for each pixel. The colour shades which are not present in the image are not allotted any row in Table 1 and hence in Table 2. The colour vectors are entered in the Table 1 in the order of their appearance in the image or in other words, as they are encountered during the scan of the

image which starts from the top left corner of the image. The population (H) in Table 1 must satisfy equation 5:

$$\sum H = m.n \quad \text{--- (5)}$$

The proposed histogram computation procedure given below finds out the colour shades available in the image and arranges them in the said format in a single scan of the complete image. Thus it does not require three scans of the complete image as in [14] neither it requires as many passes as minimum of the R,G and B frequencies as in [16]. It also simultaneously notes the positional information in a separate data structure which may further be used by different algorithms like shell clustering algorithms [17, 18], which require positional information. The histogram computing algorithm with our approach is presented below.

Algorithm 1: JND histogram computing algorithm

- i) Initialize two data structures Table 1 and Table 2. Initialize the first entry in Table 1 by the first colour vector in the image i.e. top left pixel colour vector [R,G,B] and the frequency(population) by one. Initialize the first entry in Table 2 by the top left pixel position row and column i.e. y (column)=1 and x(row)=1 and the current row index value of Table 1 i.e. 1. Also initialize a (row, column) pointer to top left corner of the image. Select a proper similarity threshold (histogram binning threshold), Θ_1 ($JND_{eye}^2 \leq \Theta_1 \leq JND_h^2$), depending on the precision of vision from fine to broad as required by the application.
- ii) Read the next pixel colour vector in scan line order.
- iii) Compare the new pixel colour vector with all the previous entries in Table 1 one by one and if found similar to any of them, then accommodate it in the respective bin. Update Table 2 by entering the current index and the current row and column values and go to step v.
- iv) If the new colour vector is not equal to any of the previously recorded colour vectors in Table 1, increment the row index of Table 1, enter the new colour vector in it, set the population to 1, make the index, row and column entry in Table 2 and go to step ii.
- v) Repeat step ii) to iv) for all the pixels in the image.
- vi) Sort Table 2 in the increasing order of the colour index.
- vii) Save the Table 1 and Table 2 for latter analysis of the histogram.

The histogram computed using JND_h^2 derived in section 2.2 as threshold, using above algorithm has miraculously reduced the colours in the natural images. The drastic reduction in number of colours in a natural image brings it to the range suitable for the machine analysis in real time. The k visually different colours in Table 1, found by the basic algorithm are further reduced using agglomeration procedure discussed in the next section.

3 Histogram Agglomeration

Agglomeration in chemical processes attributes to formation of bigger lumps from smaller particles. In the digital image segmentation, the similar pixels (in some sense) are clustered together under some similarity criteria. Thus it was inspired that the agglomeration may contribute considerably in the process of colour image segmentation. In this section, a basic agglomeration histogram processing algorithm is presented. The multidimensional histogram peak detection and thresholding are complex and time consuming tasks. The agglomeration techniques can be thought of as the powerful alternatives to the other image thresholding techniques. After the compressed histogram of a real life image is obtained using the basic JND histogram algorithm given in section 2, the agglomeration or region merging technique can further be used to reduce the number of colours by combining the smaller segments (less than 0.1% [19] of the image size) with similar coloured larger segments. To implement this scheme, an agglomeration threshold θ_2 which is slightly greater than Θ_1 is used, typically $\Theta_2 = \Theta_1 + 100$ works well here. The basic agglomeration algorithm is presented below.

Algorithm 2: Basic agglomeration algorithm

- i) Arrange the Table 1 in decreasing order of population.
- ii) Starting from the first colour in Table 1, compare the colour with the next colour in table 1.

- iii) If the population of the smaller segment is smaller than .1% and the two segments are similar using θ_2 , merge the i_{th} colour with the previous one (the first in Table 1), their populations will be added and the colour of larger population will represent the merger.
- iv) The merged entry will be removed from Table 1. This reduces number of rows in Table 1. In Table 2 the colour index to be merged is changed by the index to which it is merged.
- v) Thus the first colour in the Table 1 will be compared with every remaining colour in Table 1 and step ii is repeated if required.
- vi) Step ii, iii and iv are repeated for every colour in the Table 1.
- vii) Steps ii to v are repeated till the Table 1 does not reduce further i.e. equilibrium has reached.
- viii) Table 2 is sorted in ascending order of the colour index.

The human retina performs a low pass filtering operation following Poisson's distribution around every point on the retinal image and the neural activity initially notes and interprets the predominant or above average outputs of the retinal sensors passed to the brain via visual cortex [13]. Though we have not implemented the classical Poisson's distribution based spatial integration, the agglomeration in this work has carried out the task of low pass filtering in the colour space. The data structures given in Table 1 and Table 2 can be used to estimate parameters of FCM clustering method of image segmentation as given by the following algorithmic steps:

- i) Find out average population of colour shades in the completely agglomerated histogram.
- ii) The colour shades having their population more than the average population are the major segments in the image and the colours themselves are the respective colour segment centres.
- iii) The colours having their population below average population are the minor segments.
- iv) The initial membership value for every pixel in the image to the major segment can be computed using the Bezdek's proposition [1] as in equation 6. These memberships will be used as the initial partition for iterative clustering algorithm.

$$\mu = \frac{1}{1 + \alpha d^2} \quad \text{--- (6)}$$

Where 'd' is the Euclidean distance between the colour coefficients and the cluster seed. Euclidean norm is the most suitable norm for representing the distances in the 3-D space. Other distance measures like Manhattan, Mahalanobis and city block may be used in place of the Euclidean. However in the three dimensional RGB space the Euclidean distance appears to be the most appropriate one. Note that the colour coefficients and the cluster seeds are scaled to the interval [0,1] and α is an arbitrary constant to control the dependence of the memberships on the distances, such that $0 \leq \alpha \leq 1$. In other words the ' α ' controls fuzziness.

4 Fuzzy C-Means Clustering on JND Histogram

Fuzzy C-means (FCM) clustering algorithm [3,4,5,6,7,8,23] yields extremely good results in image region clustering and object classification. It is based on minimization of objective function. A typical objective function used is given by equation 7.

$$J(U, V) = \sum_{i=1}^c \sum_{k=1}^N \mu_{ik}^m \|x_k - v_i\|^2 \quad \text{--- (7)}$$

Where:

- N is the number of sample points and c is the number of clusters.
- x_1, x_2, \dots, x_n are n sample points, i.e. $X = \{x_k\}_{k=1}^n \subset R^q$, where q is the dimension of the desired feature space (q=3 for colour images).
- $V = \{v_1, v_2, \dots, v_c\}$ are the cluster centres.

- $U = [\mu_{ik}]$ is $c \times N$ membership matrix , where μ_{ik} is the membership value of the k^{th} input sample x_k in the i^{th} cluster. The membership values satisfy the following conditions:

$$0 \leq \mu_{ik} \leq 1, \forall i, k \quad , \quad \sum_{i=1}^c \mu_{ik} = 1, \forall k \quad \text{and} \quad 0 < \sum_{k=1}^N \mu_{ik} < N, \quad \forall i$$

The third condition above indicates that each class must have at least one sample and all the samples cannot belong to a particular class.

- m is the exponent weight factor that controls fuzziness or amount of cluster overlap. In many applications $m=2$ gives better results, although there is no fixed rule for choosing m [23]. However better results were obtained in the interval $1 < m < 3$, for segmentation of real life images in our experimentation.
- The fuzzy membership function of the k^{th} sample x_k to the i^{th} cluster is given by equation 8.

$$\mu_{ik} = \left[\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}} \right]^{-1}, \quad i = 1, 2, \dots, c \quad \text{and} \quad k = 1, 2, \dots, N \quad \text{--- (8)}$$

Where: d_{ik} is the distance of x_i and v_k , i.e. $d_{ik} = \|x_i - v_k\|$ and v_k denotes the cluster centre of the k^{th} cluster . Similarly $d_{ij} = \|x_i - v_j\|$, where $\|\cdot\|$ is an Euclidean norm.

- The algorithm iteratively updates the cluster centres using equation 9:

$$v_i = \frac{\sum_{k=1}^N \mu_{ik}^m \cdot x_k}{\sum_{k=1}^N \mu_{ik}^m}, \quad i = 1, 2, \dots, c \quad \text{--- (9)}$$

Based on the above discussion, the proposed Fuzzy C-means clustering algorithm based on agglomerated JND histogram is presented below.

Algorithm 3: Agglomerated JND histogram based Fuzzy C-means clustering

- i) Estimate the number of major segments, their cluster centres and initial partition i.e. membership values from the agglomerated colour image histogram as already described in section 3.
- ii) Compute the cluster centres, and compute the initial partition as described in Algorithm 2.
- iii) Modify the cluster centres using equation 9
- iv) Modify the partition using equation 8
- v) Compute the squared error between the current cluster centres from the respective previous cluster centres.
- vi) If squared error is less than a threshold, stop, else go to step iii.

The above fuzzy c-means algorithm yields good results but is very time consuming. It has been observed that in practice an image contains number of real life objects. The objects are enclosed by edges and segments. Primarily the edges are defined by differences in the pixel values and segments are the sets of similar pixels i.e. pixels of similar values. In other words, the probability that a segment may contain the pixels of nearly same value is high. The histogram presents the population of every such colour pixel value. Thus instead of treating every pixel of an image as an individual data if a group of similar colours is treated as data element, the speed of the FCM algorithm drastically increases. Thus taking the advantage of the recurrence of the similar pixel colour values, the convergence time can be reduced. Of course, the actual reduction in the convergence time is highly dependent on the recurrence in the image i.e. lower the number of segments the faster is the convergence.

The objective function for this FCM algorithm is the same as equation 7 but the segment centres and the membership computation equations has been modified as in equation 8 and 9. Here n is the number of rows in the histogram i.e. Table 1 after agglomeration, which represents n different colours in the image. The modified FCM procedure (Fast Fuzzy C-means Algorithm) incorporating this concept is presented below.

Algorithm 4: Proposed Fast Fuzzy C-means (FFCM) clustering Algorithm

- i) Compute the agglomerated histogram of a colour image. Let 'n' represent the number of colours in the agglomerated histogram i.e. number of rows in Table 1 and c represents the number of major segments ($c < n$). Note that $n \ll N$. H_k represent the population of the k^{th} row colour ($k=1..n$). i.e. H_k is the number of pixels having colour x_k , that represents cluster k. FCM equation 9 is modified to include histogram data and is given in equation 10.

$$v_i = \frac{\sum_{k=1}^n \mu_{ik}^m \cdot x_k \cdot H_k}{\sum_{k=1}^n \mu_{ik}^m \cdot H_k}, \quad i = 1, 2, \dots, c \quad \text{--- (10)}$$

- ii) Compute the fuzzy membership, using equation 8 with N replaced by n.
- iii) Construct the FCM partition matrix of size $c \times (n+1)$ using equation 6.
- iv) The first column of the partition matrix contains the row indices of the major segments and every column is allotted a colour out of n.
- v) Find the cluster centres using equation 10 as population and membership weighted average of the colour shades. Store the cluster centres.
- vi) Using the cluster centres and equation 8 compute the new partition i.e. membership values.
- vii) Steps 4 and 5 are repeated till the error in the respective cluster centres falls below a threshold.
- viii) The converged partition is hardened to yield the segmented image.

The reduced time complexity of this algorithm over conventional FCM is worth mentioning here. Let n be the number of segment clusters and N be the number of pixels in an image. Then one iteration of FCM computes $N \times c$ membership values and c segment centres. In order to compute a single membership value it is required to compute c Euclidian distances. Hence the computation time required for one iteration of FCM is proportional to Nc^2 , whereas for the FCM applied on JND histogram with n bins, N is replaced by n, due to binning of similar colours. The total computations required for one iteration of FCM applied on JND histogram is proportional to nc^2 . It is to be noted that the computation time required for one iteration of FCM also includes time required for computation of c segment centres. Each segment centre calculation requires $2(n-1)$ additions and n multiplications (one division operation is neglected for simplicity). Thus the computation time is drastically reduced in case of proposed FCM by the factor N/n , over conventional FCM. The segmentation results of both of these algorithms are very similar but the proposed algorithm requires much less computational efforts as discussed above. The results are presented in Figure 1.

In the above described FCM algorithms it is expected that the estimated number of segments is either exact or slightly over estimated. If the estimated number of segments is exact there is no problem at all and one may achieve quite a good segmentation. But if the estimated number of segments are slightly more than the actual then it is expected that the algorithm should find out the invalid clusters, eliminate them and again reiterate through the whole procedure till all the invalid clusters are eliminated. Thus the whole process is a massive computational task. With our approach it has been found that through out the whole experimentation on the images in BSD, an estimation of number of segments of each image is very close to the human observation, and thus the computational effort required for re-iterating through the whole process is saved.

5 Experimental Results

In this section, we demonstrate the segmentation results of the proposed algorithm on natural images from Berkeley Segmentation Database (BSD)[19]. It Contains 300 real life RGB images of different categories and of same size: 481×321 pixels. It also contains benchmark segmentation results (ground-truth database) of 1633 segmented images manually obtained from 30 human subjects. i.e. multiple ground truth hand segmentations of each image.

For each image, the quality of segmentation obtained by any algorithm can be evaluated by comparing it with many ground-truth hand segmentations, provided in BSD. One of the available ground-truth segmentations is displayed here for the sake of comparison along with each image. Each figure given below

has four image frames from left to right and from top to bottom; marked from (a) through (d) showing original image, its ground-truth segmentation, image segmented using FCM and image segmented using FFCM respectively. Note the details given in image title including image number as given in BSD, and the respective performance parameters for FCM and proposed FFCM algorithms.



Figure 1: Segmentation results: (a) Original BSD image-38092 (b) Ground-truth Image (c) FCM Segmented Image (PRI=0.83,PSNR=32.71,nsegs=14,Time=235 Sec) (d) FFCM Segmented Image (PRI=0.83, PSNR=31.44,nsegs=13,Time=7.2 sec)

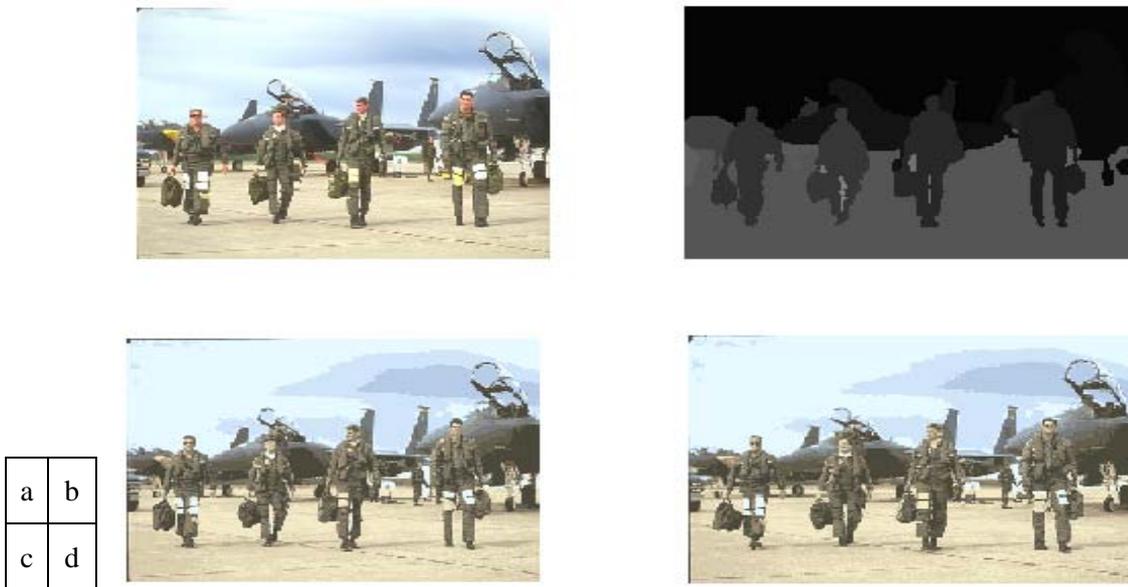


Figure 2: Segmentation results: (a) Original BSD image-245051 (b) Ground-truth Image (c) FCM Segmented Image (PRI=0.83,PSNR=34.49,nsegs=17,Time=237 Sec) (d) FFCM Segmented Image (PRI=0.85, PSNR=29.62,nsegs=14,Time=7.7 sec)

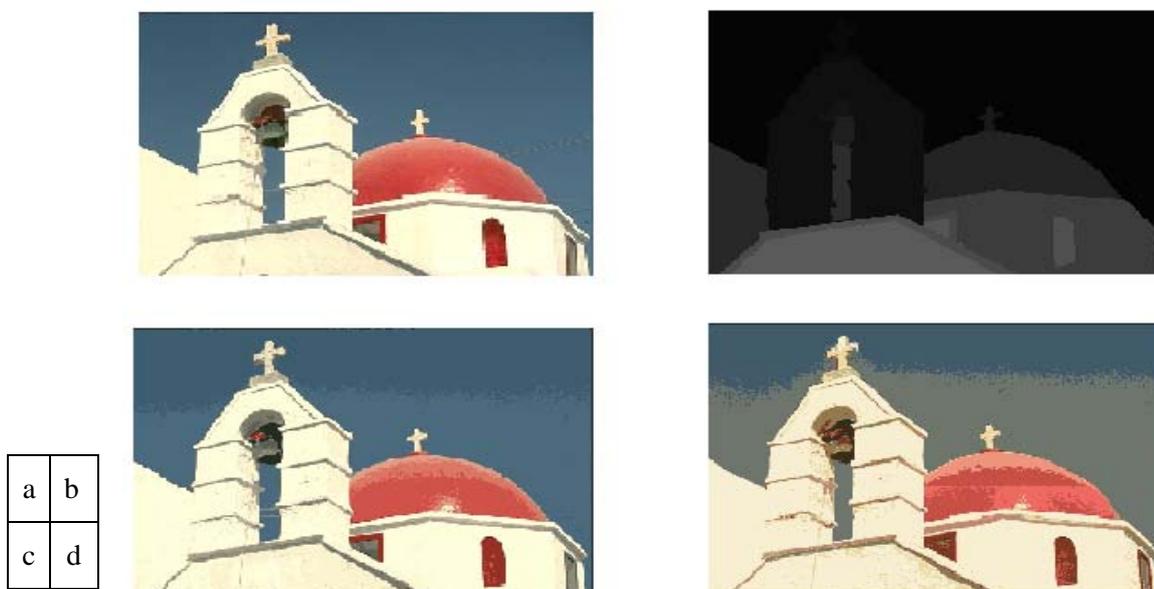


Figure 3: Segmentation results: (a) Original BSD image-118035 (b) Ground-truth Image (c) FCM Segmented Image (PRI=0.79,PSNR=36.22,nsegs=15,Time=226 Sec) (d) FFCM Segmented Image (PRI=0.78, PSNR=32.55,nsegs=14,Time=9.7 sec)

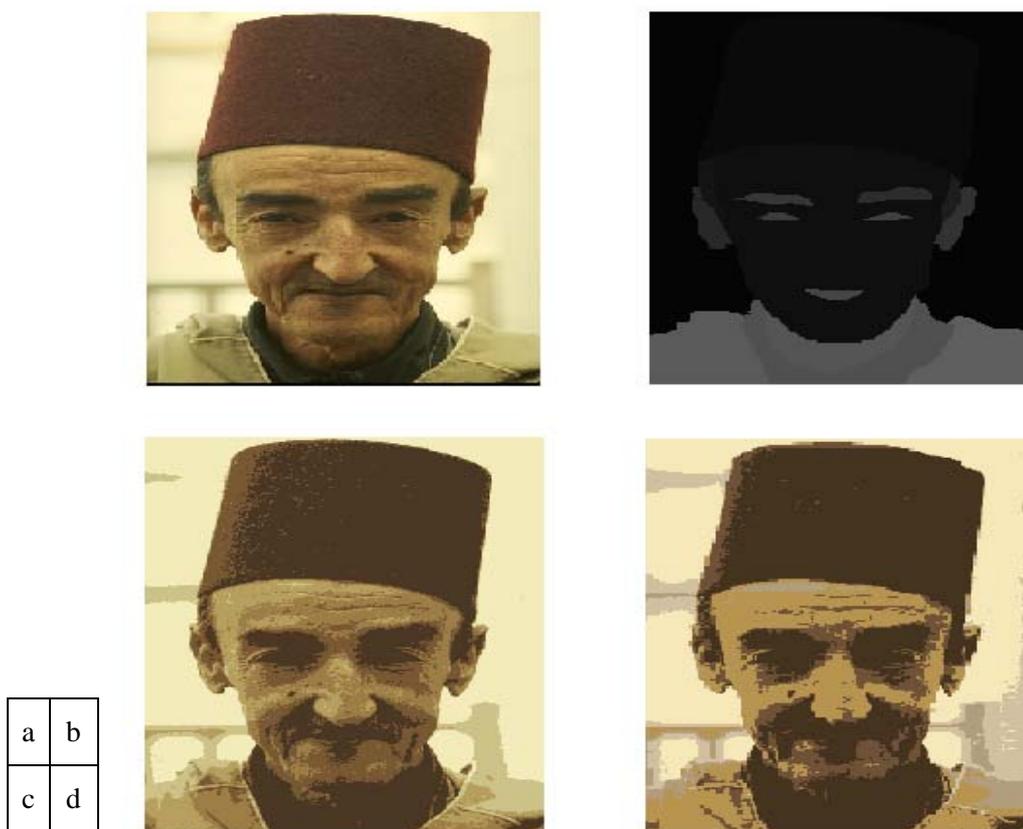


Figure 4: Segmentation results: (a) Original BSD image-118035 (b) Ground-truth Image (c) FCM Segmented Image (PRI=0.81,PSNR=33.34,nsegs=6,Time=43.1 Sec) (d) FFCM Segmented Image (PRI=0.77, PSNR=35.09,nsegs=6,Time=6.1 sec)

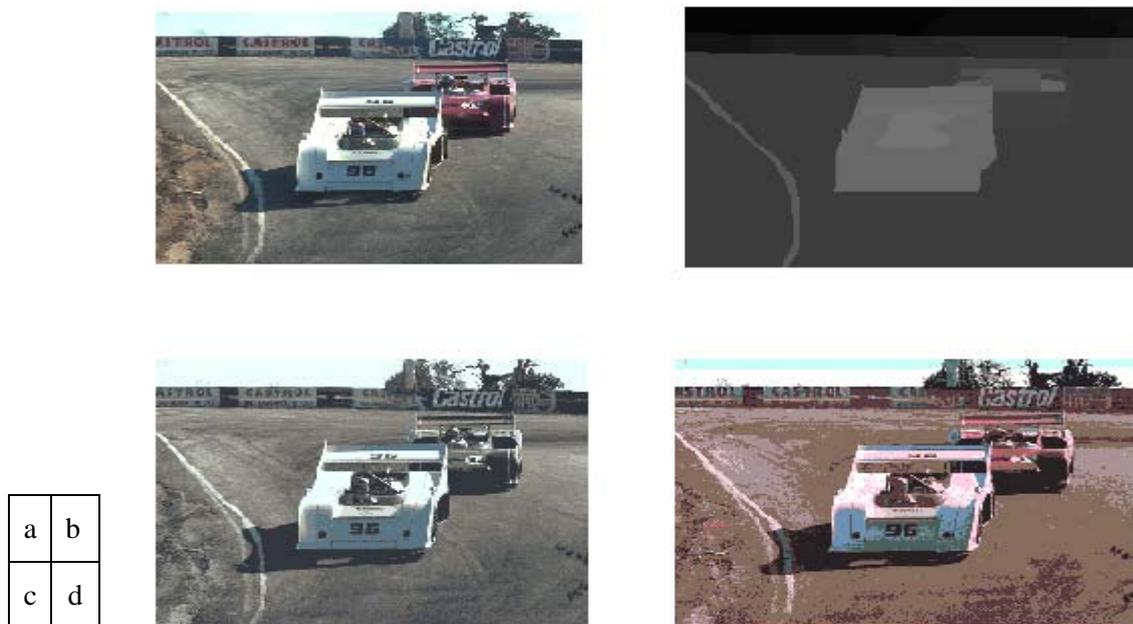


Figure 5: Segmentation results: (a) Original BSD image-21077 (b) Ground-truth Image (c) FCM Segmented Image (PRI=0.64,PSNR=34,nsegs=13,Time=153.8 Sec) (d) FFCM Segmented Image (PRI=0.71, PSNR=31.09,nsegs=13,Time=7.5 Sec)



Figure 6: Segmentation results: (a) Original BSD image-134008 (b) Ground-truth Image (c) FCM Segmented Image (PRI=0.41, PSNR=36.04,nsegs=6,Time=60 Sec) (d) FFCM Segmented Image (PRI=0.55, PSNR=29.66,nsegs=5,Time=5.8 Sec)

The results of proposed segmentation algorithm (FFCM) are presented above in Figure 1 through Figure 6, and its effectiveness is compared with conventional FCM algorithm using two quantitative measures, namely, the Probabilistic Rand Index (PRI)[21,24] and Peak signal to Noise Ratio (PSNR). The other parameters of importance, namely: number of segments found (nsegs) and time required for segmentation of an image is also provided along with each segmentation result.

PRI Counts the fraction of pairs of pixels whose labeling are consistent between the computed segmentation and the ground truth, averaging across multiple ground truth segmentations to account for variation in human perception. This measure takes the values in the interval [0,1]; more is better. The probabilistic rand index (PRI) is defined mathematically as given by equation 11:

$$PRI[(S_{test}, \{S_k\})] = \frac{1}{\binom{N}{2}} \sum_{\substack{i,j \\ i < j}} [c_{ij} \cdot p_{ij} + (1 - c_{ij})(1 - p_{ij})] \quad (11)$$

Where image $X = \{x_1, x_2, \dots, x_N\}$, N is the number of pixels, $\{S_k\}$ is the ground truth set $\{S_1, S_2, \dots, S_K\}$ for image X and S_{test} is the segmented image to be evaluated against set $\{S_k\}$. The PR models label relationships for each pixel pair (x_i, x_j) , where each human segmenter provides information (c_{ij}) about each pair of pixels as to whether the pair belongs to the same group or belongs to different groups, p_{ij} is the mean pixel pair relationship among the ground truth images or the ground truth probability that $(l_i = l_j)$.

We will consider the segmentation ‘good’ if for any pair of pixels x_i, x_j we would like the labels of those pixels l_i^{Stest}, l_j^{Stest} to be the same in the test segmentation if the labels l_i^{Sk}, l_j^{Sk} were the same in the ground truth segmentations, and vice versa.

PSNR represents region homogeneity of the final partitioning. The higher the value of PSNR the better is segmentation[22]. The PSNR measure between the image X and the first order approximation based on the segmentation result S is calculated by equation 12:

$$PSNR(X, S) = 10 \cdot \log_{10} \left(\frac{255^2 \text{ rows} \cdot \text{columns} \cdot \text{channels}}{\sum_i \sum_j \sum_k [X(i,j,k) - S(i,j,k)]^2} \right) \quad (12)$$

The algorithm is applied to whole database of 300 images. The average PRI and PSNR values of all the images for different methods are given in Table 3. The quantitative comparison as given in Table 3 and the qualitative (visual) comparison presented in Figures 1 through Figure 6, clearly demonstrate the superiority of proposed algorithm over conventional FCM in terms of time required. Also it is to be noted that the time given in column 6 of Table 3, includes the time required to find initial parameters for FCM/FFCM.

We have performed number of experiments with different values for thresholds Θ_1 and Θ_2 . The best results are found at $\Theta_1=2400$ (Algorithm 1) and $\Theta_2=2500$ (Algorithm 2), both of which are very close to JND_h value of 2567, derived in section 2.2. The results below are based on JND histogram obtained using these values of thresholds.

Segmentation Method	Average number of Segments identified on BSD	PRI	PSNR	Total Time* for Segmentation of 300 BSD images
FCM algorithm ($m=1.1, \alpha=0.8$)	13.72	0.7029	33.76	14.19 Hrs
FFCM ($m=1.1, \alpha=0.8$), based on JND Colour Histogram	10.16	0.7081	31.34	0.81 Hrs

Table 3: Average Performance on BSD (*On AMD Athlon 1.61 GHz processor, 1GB RAM and MATLAB 7 running on Windows XP)

6 Conclusions

Based on the experimentation carried out on 300 BSD images (and its 1633 ground-truth segmentations) and the subsequent results presented in section 5, it is concluded that:

- The initialization of FCM parameters, i.e. number of clusters and cluster seeds, using JND histogram approach, is an innovative method proposed here that works very well with natural images.
- It is also to be noted that the best values found empirically, for basic histogram binning threshold Θ_1 is 2400, and agglomeration threshold Θ_2 is 2500. These values are very close to $JND_h=2567$; which approves our derivation for JND_h given in section 2.2.
- The performance of conventional FCM algorithm and the proposed FFCM algorithm are more or less same in terms of PRI and PSNR. Both the algorithms perform very well in terms of PSNR, which is a measure of region homogeneity. From Table 3 it can be noted that there is a significant time improvement (roughly 18 times on an average), with the proposed FFCM approach.
- The number of segments on an average are reduced by about 20% in FFCM in comparison with FCM, without adversely affecting the visual quality or the performance parameters.
- Although the linked data structure (Table 1 and Table 2) adds to time and space complexity, it is quite useful for segment level matching for CBIR algorithms and image annotation applied on real life colour images.

References

- [1] C. V. Jawahar and A. K. Ray, "Fuzzy statistics of digital images", *IEEE Signal Processing Letters*, Vol.3, No.8, August 1996, pp. 225-227.
- [2] C. V. Jawahar and A. K. Ray, "Techniques and Applications of Fuzzy Statistics in Digital Image Analysis", *Fuzzy Theory Systems: Techniques and Applications*, Ed. C.T. Leondes, Academic Press, 12, 1999, pp 759-778.
- [3] Tinku Acharya and Ajoy K. Ray, "Image Processing –Principles and Applications", *Wiley Interscience.*, 2005,pp. 219-223
- [4] J. C. Bezdek, *Pattern Recognition With Fuzzy Objective Function Algorithms*, *Plenum Press*, New York, 1981
- [5] A. K. Jain, M. N. Musty and P. J. Flynn, "Data Clustering: A Review", *ACM Computing Survey*, Vol. 31. No 3, pp. 264-323, 1999.
- [6] L. Kaufman and P. Rousseeuw, "Finding Groups in Data, An Introduction to Cluster Analysis", *John Wiley & Sons*, 1990.
- [7] A. Hafiane, B. Zavidovique, and S. Chaudhuri, "A Modified FCM With Optimal Peano Scans For Image Segmentation", *Proceedings of ICIP 2005*, Vol. III, pp 840-843.
- [8] Songcan Chen and Daoqiang Zhang, "Robust Image Segmentation Using FCM With Spatial Constraints Based on New Kernel-Induced Distance Measure", *IEEE Transactions On Systems, Man, and Cybernetics, Part B*, Vol. 34, issue 4, August 2004, pp 1907-1916.
- [9] Zhang Yang, Fu-Lai Chung, Wang Shitong, "Robust Fuzzy Clustering-based Image Segmentation", *Applied Soft Computing Journal*, vol. 9, issue 1, Jan. 2009, pp 80-84.
- [10] Edward Y.Chang and Beita Li, Chen Li, "Toward Perception-based Image Retrieval", *IEEE Workshop on Content-based Access of Image and Video Libraries*, 2000.
- [11] K. M. Bhurchandi, P. M. Nawghare, A. K. Ray, 'An analytical approach for sampling the RGB colour space considering' limitations of human vision and its application to colour image analysis', *Proceedings of ICVGIP 2000, Bangalore*, pp.44-49.
- [12] A. C. Guyton, "A text book of medical Physiology", *W.B.Saunders company*, Philadelphia,1976, pp.784-824.

- [13] Gaurav Sharma, "Digital colour imaging", *IEEE Transactions on Image Processing*, Vol. 6, No.7, July1997, pp.901-932.
- [14] Young-Chang Chang and John F. Reid, "RGB calibration for colour image analysis in machine vision", *IEEE Transactions on Image Processing*, Vol. 5, No.10, October 1996, pp. 1414-1422.
- [15] A. Moghaddamzadeh and N. Bourbakis, "A fuzzy region growing approach for segmentation of colour images", *Pergamon, Pattern Recognition*, Vol.30,No.6,1997, pp.867-881.
- [16] Sang Ho Park, Il Dong Yun and Sang Uk Lee, "Colour image segmentation based on 3-D clustering: morphological approach", *Pergamon, Pattern Recognition*, Vol.44, No.8, 1998, pp. 1061-1076.
- [17] Liang-Kai Huang and Mao-Jiun J.Wang, "Image thresholding by minimizing the measures of fuzziness", *Pergamon,Pattern Recognition*, Vol.28,No.1,1995,pp.41-51.
- [18] Raghu Krishnapuram, Hichem Frigui and olfa Nasraoui, "Fuzzy possibilistic shell clustering Algorithms and their application to boundary detection and surface approximation- part I", *IEEE Transactions on Fuzzy Systems*, Vol.3,No.1, February1995 ,pp.29 -43.
- [19] Raghu Krishnapuram, Hichem Frigui and Olfa Nasraoui, "Fuzzy possibilistic shell clustering Algorithms and their application to boundary detection and surface approximation- part II", *IEEE Transactions on Fuzzy Systems*, Vol.3,No.1, February1995 ,pp.44-60.
- [20] D. Martin, C. Fowlkes, D. Tal, J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics", *Proceedings of IEEE International Conference on Computer Vision, 2001*, pp.416 –423
- [21] R. Unnikrishnan, M. Hebert, "Measures of Similarity", *IEEE Workshop on Computer Vision Applications, 2005*, pp. 394–400.
- [22] Milind M. Mushrif, Ajoy K. Ray,"A-IFS Histon Based Multithresholding Algorithm for Colour Image Segmentation" ,*IEEE Signal Processing Letters*, vol. 16, No. 3, 2009.
- [23] Nikhil R. Pal and James C. Bezdek, "On Cluster Validity for the Fuzzy C-Means Model", *IEEE Transaction on Fuzzy Systems*, Vol. 3 No.3 August 1995, pp. 370–379.
- [24] Weiling Cai, Songcan Chen and Daoqiang Zhang, "Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation", *Pattern Recognition*, Vol. 40 , Issue 3, March 2007, pp. 825-838.
- [25] Martin Tabakov, "A Fuzzy Clustering Technique for Medical Image Segmentation", *International Symposium on Evolving Fuzzy Systems*, September, 2006, pp. 118-122.
- [26] Yongqiang Zhao Minglu Li, "Modified Fuzzy C-Means Algorithm for Segmentation of MRI", *Proceedings of the Fifth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA'03)*.
- [27] Jianchao Fan, Min Han and Jun Wang, "Single point iterative weighted fuzzy C-means clustering algorithm for remote sensing image segmentation", *Elsevier Pattern Recognition*, Vol.42, Issue 11, Nov. 2009, pp. 2527-2540