ISSN 1512-1232

FUSION APPROACH FOR REGION BASED IMAGE SEGMENTATION USING K-MEANS CLUSTERING

Dr.G.Samuel Varaprasad Raju[1], G.Syam Prasad[2], Y.Srikanth[3]

[1] Professor in Scholl of Distance Education,AU,Vizag

[2]Associate Professor in CSE Dept,sriviveka institute of technology , Vijayawada

[3]B.Tech, CSE,sriviveka institute of technology , Vijayawada

ysrikanth557@yahoo.com

Abstract

This paper presents a new, simple and efficient Region based segmentation approach, based on a fusion procedure which aims at combining several segmentation maps associated to simpler partition models in order to finally get a more reliable, accurate and a non-overlapped image result. The main objective of the paper is to get a non-overlapping and a reliable output by using k-mean algorithm. The different colorspaces are to be fused in our application by the simple (K-means based) clustering technique on an input image. The optimized range for k-means clustering values is obtained by performing genetic algorithm. Image segmentation for six color spaces are performed by kmeans. The k-means algorithm is an iterative technique that is used to partition an image into clusters. The obtained output remains simple to implement, fast, general enough to be applied to various computer vision applications (e.g., motion detection and segmentation). The result aims at developing an accurate and more reliable image which can be used in locating tumors, measure tissue volume, face recognition, finger print recognition and in locating an object clearly from a satellite image and in more.

Key words*: color spaces, fusion of segmentations,k -means clustering, textured image segmentation*

1 . INTRODUCTION

IMAGE segmentation is a classic inverse problem which consists of achieving a compact region based description of the image scene by decomposing it into meaningful or spatially coherent regions sharing similar attributes. This low-level vision task is often the preliminary (and also crucial) step in many video and computer vision applications, such as object localization or recognition, data compression, tracking, image retrieval, or understanding. Because of its simplicity and efficiency, clustering approaches were one of the first techniques used for the segmentation of (textured) natural images [1]. After the selection and the extraction of the image features [usually based on color and/or texture and computed on (possibly) overlapping small windows centered around the pixel to be classified],the feature samples, handled as vectors, are grouped together in compact but well-separated clusters corresponding to each class of the image. The set of connected pixels belonging to each estimated class thus defined the different regions of the scene. The method known as k-means [3] (or Lloyd's algorithm).

2. IMAGE SEGMENTATION

In computer vision, **segmentation** refers to the process of partitioning a digital image into multiple regions (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images.

ISSN 1512-1232

The result of image segmentation is a set of regions that collectively cover the entire image, or a set of contours extracted from the image . Each of the pixels in a region are similar with respect to some characteristic or computed property, such as color, intensity, or texture. Adjacent regions are significantly different with respect to the same characteristic(s).

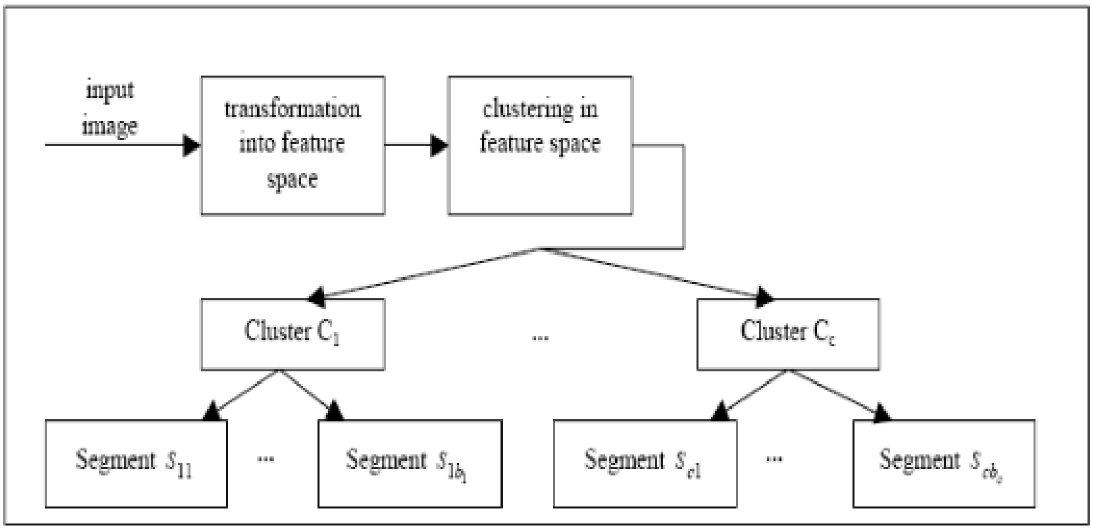


Fig. 1. Flow-chart of an image segmentation method

A. Region Based Segmentation

Region-based segmentation methods attempt to partition or group regions according to common image properties. These image properties consist of

1. Intensity values from original images, or computed values based on an image operator
2. Textures or patterns that are unique to each type of region
3. Spectral profiles that provide multidimensional image data

Elaborate systems may use a combination of these properties to segment images, while simpler systems may be restricted to a minimal set on properties depending of the type of data available.

3. COLOUR SPACES

A colour space is a method by which we can specify, create and visualise colour. As humans, we may define a colour by its attributes of brightness, hue and colourfulness. A computer may describe a colour using the amounts of red, green and blue phosphor emission required to match a colour. A printing press may produce a specific colour in terms of the reflectance and absorbance of cyan, magenta, yellow and black inks on the printing paper.

A colour is thus usually specified using three co-ordinates, or parameters. These parameters describe the position of the colour within the colour space being used. They do not tell us what the colour is, that depends on what colour space is being used.

A. Different colour spaces

The image first considered is a color image; color image is taken because this image is further divided into different color spaces. Different colors provide different properties in special. A color image gives detailed information about the image and an attractive way of producing an image. Thus a color image is considered, it can a JPEG, BMP image.The color space may be device dependent and device independent,according to CIE standard six color spaces namely rgb,yiq,xyz,hsv,luv,lab are considered.

B. RGB (Red Green Blue)

This is an additive color system based on tri-chromatic theory. RGB is easy to implement but non–linear with visual perception. It is device dependent and specification of colors is semiintuitive. RGB is very common, being used in virtually every computer system as well as television, video etc.

C. Hue Saturation Value (Travis)

These are the RGB-HSV conversions given by Travis. To convert from RGB to HSV (assuming normalized RGB values) first find the maximum and minimum values from the RGB triplet. s= (max-min)/max and Value is V=maxThe Hue, H is then calculated as follows. First calculate R'G'B'. Conditions is given by,

R'=(max-R)/(max-min)

G'=(max-G)/(max-min)

B'=(max-B)/(max-min)

Hue, H, is then converted to degrees by multiplying by 60 giving HSV[18] with S and V between 0 and 1 and H between 0 and 360.

D. Y’I’Q’ Color Space

Y’ is similar to perceived luminance, I’ and Q’ carry color information and some luminance information and are derived by rotating the U’V’ vector formed by color coding .The Y’ signal usually has 4.2 MHz bandwidth in a 525 line system. Originally the I’ and Q’ signals were to have different bandwidths (0.5 and 1.5 MHz) but they now commonly have the same bandwidth (1 MHz). The system white point is Illuminant C, the chromaticity co-ordinates are:-

R: xr=0.67 yr=0.33

G: xg=0.21 yg=0.71

B: xb=0.14 yb=0.08

white: xn=0.310063 ;yn=0.316158

E. XYZ Color Space

Conversion of RGB image pixel values to the CIE XYZ tristimulus values of the color displayed on the CRT can be achieved using a two stage process. Calculate the relationship between input image pixel values and displayed intensity. This relationship is the transfer function, it is simplified to gamma. The transfer functions will usually differ for each channel so are best measured independently. The second stage is to transform between the displayed red, green and blue to the CIE[25] tri stimulus values. This is most easily performed using a matrix transform of the following form

$$\begin{vmatrix} X \\ Y \\ Z \end{vmatrix} = \begin{vmatrix} X_r & X_g & X_b \\ Y_r & Y_g & Y_b \\ Z_r & Z_g & Z_b \end{vmatrix} * \begin{vmatrix} R \\ G \\ B \end{vmatrix} \quad \text{.....(1)}$$

Where X, Y,Z are the desired CIE tri-stimulus values, R,G,B are the displayed RGB values obtained from the transfer functions and the 3x3 matrix is the measured CIE tri-stimulus values for CRT.

F. LAB Color Space

This is based directly on CIE XYZ (1931) and is another attempt to linearise the perceptibility of unit vector color differences. Again, it is non-linear, and the conversions are still reversible. Coloring information [5] is referred to the color of the white point of the system, subscript n. The nonlinear relationships for L* a* and b* are the same as for CIE LUV

$$L^* = \begin{cases} 116(\frac{Y}{Y_n})^{\frac{1}{3}} - 16 & \text{if } \frac{Y}{Yn} > 0.008856 \\ 903.3(\frac{Y}{Y_n}) & \text{if } \frac{Y}{Yn} \leq 0.008856 \end{cases} \quad \text{.....(2)}$$

Again, L_ scales from 0 to 100.Again, there are polar parameters that more closely match the visual experience of colors. C*=(a*2 +b*2)0.5 ; hab = arctan (b*/a*) Hue is an angle in four quadrants, and there is no saturation term in this system. When determining CIEL*a*b* or

ISSN 1512-1232

CIEL*u*v* values for CRT displayed colors it is usual to used the CRT's white point as the reference white.

G. *L*U*V* Color Space*

It is used to linearise the perceptibility of unit vector color differences. It is a non-linear color space, but the conversions are reversible. Coloring information is centered on the color of the white point of the system, subscript n, (D65 in most TV systems). The non-linear relationship for Y* is intended to mimic the logarithmic response of the eye.

$$L^* = \begin{cases} 116(\frac{Y}{Y_n})^{\frac{1}{3}} - 16 & \text{if } \frac{Y}{Yn} > 0.008856 \\ 903.3(\frac{Y}{Y_n}) & \text{if } \frac{Y}{Yn} \leq 0.008856 \end{cases}$$

$$u^* = 13(L^*)(u' - u'_n)$$
$$v^* = 13(L^*)(v' - v'_n) \qquad \text{….. (3)}$$

4. SEGMENTATION IN DIFFERENT COLOUR SPACES

The initial segmentation maps which will then be fused together by fusion framework are given, in this application, by a K-means[3][6] clustering technique, applied on an input image expressed by different color spaces, and using as simple cues (i.e., as input multidimensional feature descriptor) the set of values of the re-quantized color histogram (with equidistant binning) estimated around the pixel to be classified. In this application, this local histogram is equally re-quantized (for each of the threecolor channels) in a Nb=5*5*5=125 bins descriptor, computed on an overlapping squared fixed-size (Nw=7) neighborhood centered around the pixel to be classified. This estimation can be quickly computed by using a more coarsely requantized color space and then computed the bin index that represents each re-quantized color.

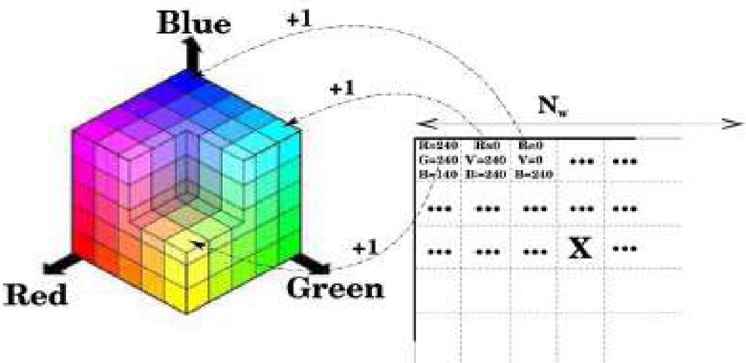


Fig 2. Estimation of each pixel x

Mathematically, let b(x) €{0,….Nb-1 denote the bin index associated with the color vector y(x) at pixel location (lying on a pixel grid) and N(x) be the set of pixel locations X within the squared neighborhood region (of fixed-size Nw*Nw) centered at pixel location X (in which local color information will be gathered).An estimate of h(x)={h(n;x)}n=0 ….Nb-1 of 125 bins descriptor, characterizing the color distribution[4] for each pixel to be classified, is given by the following standard bin counting procedure:

$$h(n;x) = k \sum_{u \in N(x)} \delta\,[b(u)-n] \qquad \text{…..(4)}$$

ISSN 1512-1232

Where Kronecker delta function and is a k=1/(Nw) 2 normalization constant ensuring is a normalization constant ensuring Σn=0.Algorithm I. Estimation, for each pixel, of the bins descriptor. Estimation of the Nb=q3 bins descriptor Nx set of pixel locations x within the NwNw neighborhood region centered at h[]Bins descriptor: Array of Nb floats (h[0],h[1],......h[Nb-1]). [.] Integer part For each pixel x□Nx with color value Rx,Gx,Bx

K→ q2.[q.Rx/256]+q.[q.Gx/256]+[q.Bx/256]
h[k] →h[k]+1/(Nw)^2

Finally, these (125-bin) descriptors are grouped together into different clusters (corresponding to each class of the image) by the classical K -means algorithm with the classical Euclidean distance. This simple segmentation strategy of the input image into K1 classes is repeated for different color spaces which can be viewed as different image channels provided by various sensors or captors (or as a multichannel filtering where the channels are represented by the different color spaces). In our application, we use Ns segmentations provided by the Ns=6 color spaces namely RGB,HSV,YIQ,XYZ,LAB,LUV color spaces.

6. K-MEANS CLUSTERING

K-means (MacQueen, 1967) is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids shoud be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more. Finally, this algorithm aims at minimizing an *objective function*, in this case a squared error function. The objective function

$$J = \sum_{j=1}^{k}\sum_{i=1}^{n}\left\|x_i^{(j)} - c_j\right\|^2,$$

where $\left\|x_i^{(j)} - c_j\right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre c_j, is an indicator of the distance of the *n* data points from their respective cluster centres.

The algorithm is composed of the following steps:

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2. Assign each object to the group that has the closest centroid.
3. When all objects have been assigned, recalculate the positions of the K centroids.
4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculate.

Although it can be proved that the procedure will always terminate, the k means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centres. The k-means algorithm can be run multiple times to reduce this effect. K-means is a simple algorithm that has been adapted to many problem domains. As we are going to see, it is a good candidate for extension to work with fuzzy feature vectors.

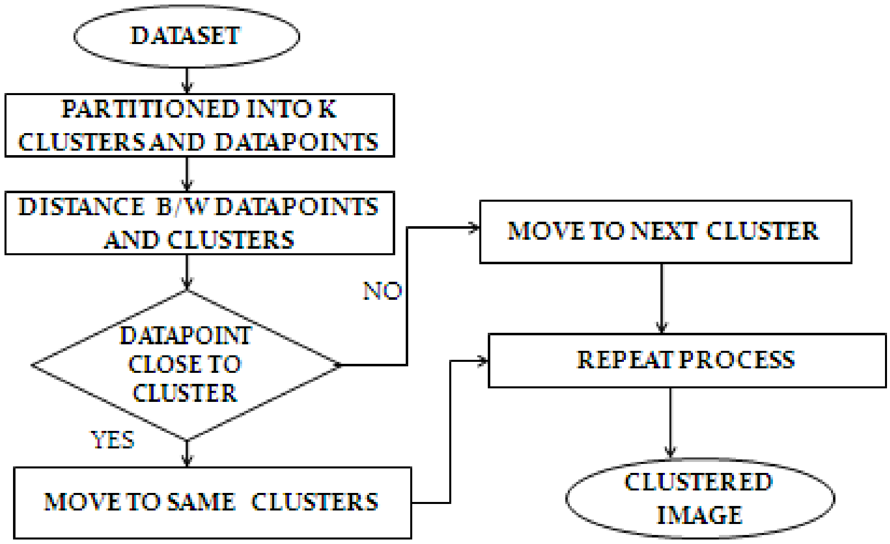


Fig 3. Flow chart for k-Means Algorithm

7. PROCESS OF GENERIC ALGORITHM

Step 1: The algorithm usually starts from a population of randomly generated individuals and happens in generations.

Step 2: In each generation, the fitness of every individual in the population is evaluated. Step 3: Multiple individuals are randomly selected from the current population and modified to form a new population.

Step 4: The new population is then used in the next iteration of the algorithm.

Step 5: The algorithm terminates when either a maximum number of generations has been produced, or a satisfactory

fitness level has been reached for the population.

Step 6: Genetic algorithms find application in bioinformatics, phylogenetics, engineering and manufacturing

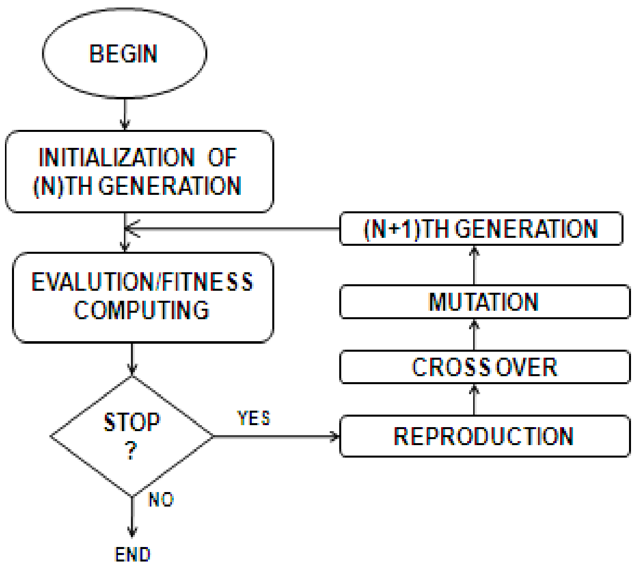


Fig 4. Flow chart for Genetic Algorithm

ISSN 1512-1232

Color Space Conversions:

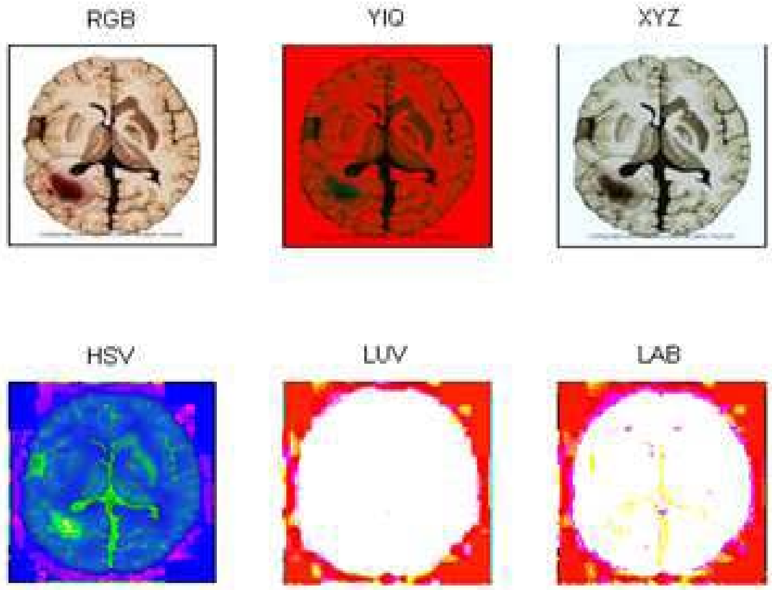


FIG Segmented Images Using k-Means

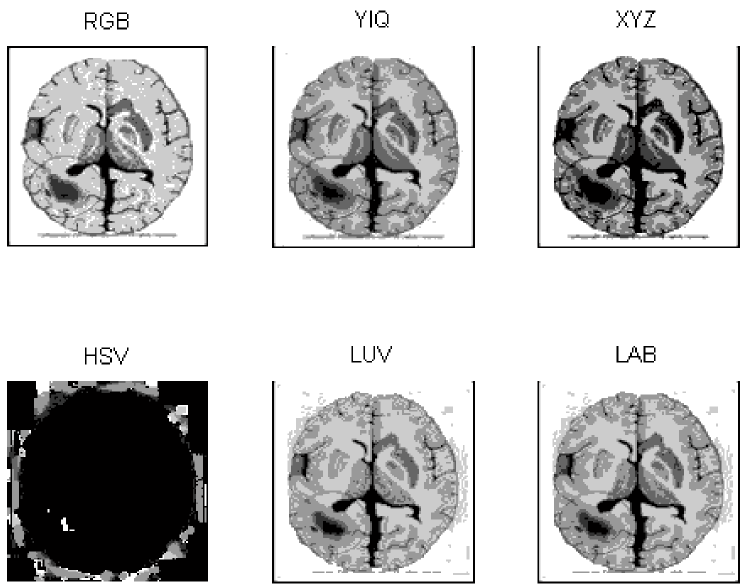
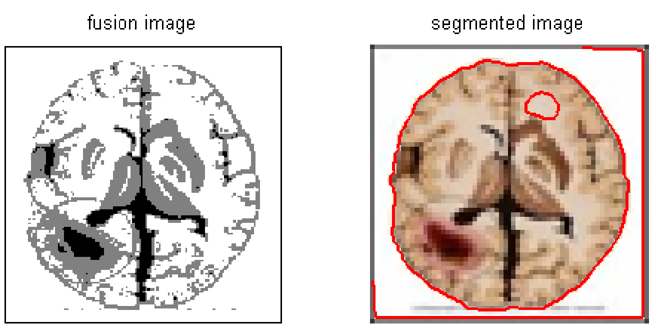


Fig 6. Segmented images using k-means

Fused Images:



ISSN 1512-1232

Fig 7. fused image using k-means

PERFORMANCE MEASURE

Table 1. Average performance of algorithm for several values of its internal parameters

Parameter	Boundary	RI	VOI	GCE
RGB(proposed) (existing)	4.4713 9.3065	0.66438 0.80910	3.4926 2.1282	0.49276 0.17975
YIQ(proposed) (existing	4.5963 11.1331	0.64427 0.79055	3.3973 2.2627	0.47207 0.26257
XYZ(proposed) (existing	3.5852 11.5585	0.69099 0.80683	3.6899 2.0082	0.56648 0.25385
HSV(proposed) (existing)	4.3259 9.8310	0.68278 0.79904	3.2855 2.4031	0.47484 0.16182
LUV(proposed) (existing)	4.1055 10.8777	0.61257 0.74751	3.2714 3.0254	0.42793 0.33568
LAB(proposed) (existing)	5.444 9.3800	0.61751 0.80549	3.2931 2.4849	0.44131 0.22742
FUSION(proposed) (existing)	6.6523 10.2006	0.57372 0.83375	3.2766 1.6808	0.4151 0.13678

Where GCE- Global consistency Measure ,BDE-Boundary Displacement Error

ISSN 1512-1232

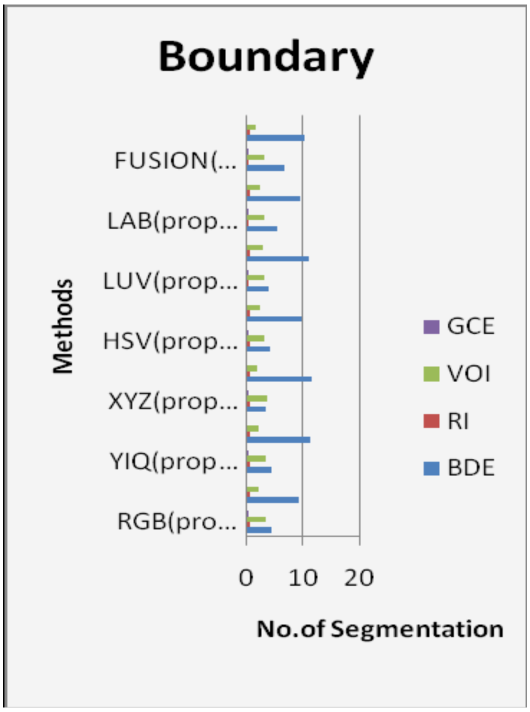


Fig.8a. characteristics of boundary values

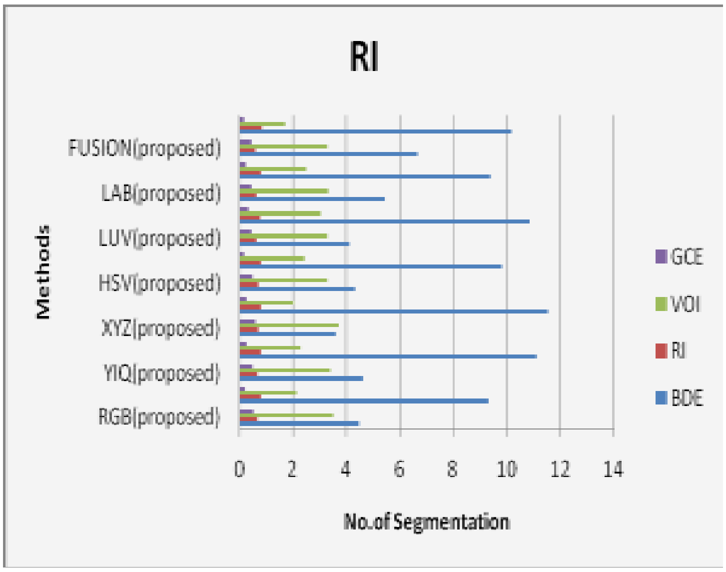


Fig.8c. characteristics of RI values

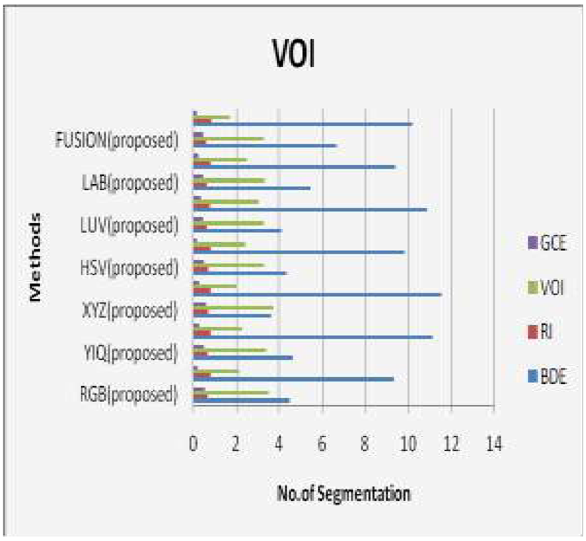


Fig.8b. characteristics of VOI values

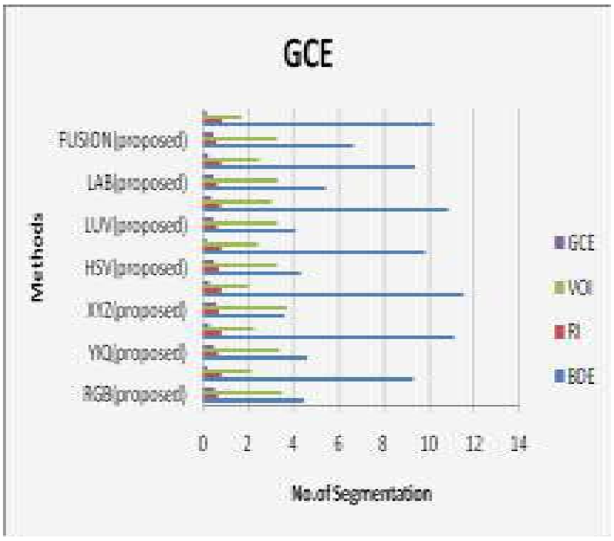


Fig.8d. characteristics of GCE values

ISSN 1512-1232

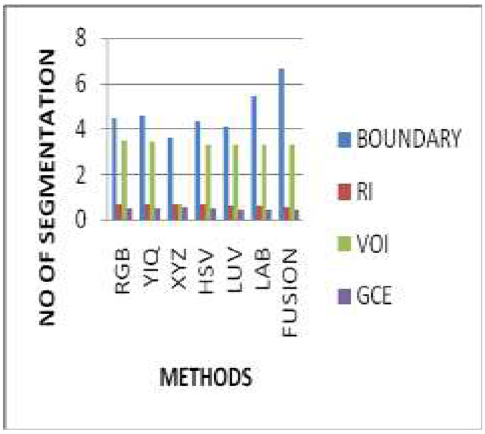


Fig 8e. Total Performance Analysis

CONCLUSION

A new segmented image using genetic algorithm is fused to get more reliable and accurate result. This fusion framework remains simple, fast, easily parallelizable, general enough to be applied to various computer vision applications, and performs competitively among the recently reported state-of-the-art segmentation methods.This novel scheme can be used in the image processing and computer vision community, and has widespread applications in many application areas ranging.

REFERENCES

[1] S. P. Lloyd, ―Least squares quantization in PCM, *IEEE Trans. Inf.Theory*, vol. IT-28, no. 2, pp. 129–136, Mar.1982.
[2] D. Comaniciu and P. Meer, ―Mean shift: A robustapproach toward feature space analysis, *IEEE Trans.Pattern Anal. Mach. Intell.*, vol.24, no. 5, pp. 603–619, May 2002.
[3] J. Shi and J. Malik, ―Normalized cuts and image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug.2000.
[4] S. Zhu and A. Yuille, ―Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 9, pp. 884–900, Sep.1996.
[5] M. Mignotte, C. Collet, P. Pérez, and P. Bouthemy, ―Sonar image segmentation using a hierarchical MRF model, *IEEE Trans. Image Process.*, vol. 9, no. 7, pp. 1216–1231, Jul. 2000. [6] F. Destrempes, J.-F. Angers, and M. Mignotte, ―Fusion of hidden Markov random field models and its Bayesian estimation, *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 2920–2935, Oct. 2006.
[7] J.-P. Braquelaire and L. Brun, ―Comparison and optimization of methods of color image quantization, *IEEE Trans. Image Process.*, vol. 6, no. 7, pp. 1048–1952, Jul. 1997.
[8] H. Stokman and T. Gevers, ―Selection and fusion of color models for image feature detection, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 371–381, Mar. 2007.
[9] R. Unnikrishnan, C. Pantofaru, and M. Hebert, ―A measure for objective evaluation of image segmentation algorithms, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshop on Empirical Evaluation Methods in Computer Vision*, Jun. 2005, vol. 3, pp.34–41.
[10] S. Banks, *Signal Processing, Image Processing and Pattern Recognition eng*lewood Cliffs, NJ: Prentice-Hall, 1990.
[11] P. Berkhin, ―Survey of clustering data mining techniques, AccrueSoftware, San Jose, CA, 2002.

ISSN 1512-1232

[12] J. Besag, ―On the statistical analysis of dirty pictures,‖ *J. Roy. Statist.Soc. B*, vol. 48, pp. 259–302, 1986.
[13] P. Felzenszwalb and D. Huttenlocher, ―Efficient graph-based image segmentation,‖ *Int. J. Comput. Vis.*, vol. 59, pp. 167–181, 2004.
[14] M. Mignotte, C. Collet, P. Pérez, and P. Bouthemy, ―Three-class Markovian segmentation of high resolution sonar images,‖ *Comput. Vis. Image Understand.*, vol. 76, no. 3, pp. 191–204, 1999.
[15] Z. Kato, T. C. Pong, and G. Q. Song, ―Unsupervised segmentation of color textured images using a multi-layerMRF model,‖ in *Proc. Int. Conf. Image Processing*, Barcelona, Spain, Sep. 2003, pp. 961–964.
[16] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, ―Color-based probabilistic tracking,‖ in *Proc. Eur. Conf.Computer Vision*, Copenhagen, Denmark, Jun. 2002, pp. 661–675.
[17] J. B. Martinkauppi, M. N. Soriano, and M. H. Laaksonen, ―Behavior of skin color under varying illumination seen by different cameras at different color spaces,‖ in *Proc. SPIE, Machine Vision Applicatios in Industrial Inspection IX*, San Jose, CA, Jan. 2001, pp. 102–113.
[18] Z. Kato, ―A Markov random field image segmentation model for color textured images,‖ *Image Vis. Comput.*, vol. 24, no. 10, pp. 1103–1114, 2006.
[19] E. Maggio and A. Cavallaro, ―Multi-part target representation for color tracking,‖ in *Proc. Int. Conf. ImageProcessing*, Italy, Genova, Sep.2005, pp. 729–732 [20] M. Meila, ―Comparing clusterings—An axiomatic view,‖ in *Proc. 22nd Int. Conf. Machine Learning*, 2005, pp. 577–584.
[21] D. Martin, C. Fowlkes, D. Tal, and J. Malik, ―A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,‖ in *Proc. 8th Int. Conf. Computer Vision*, Jul. 2001, vol. 2, pp. 416–423.
[22] J. Freixenet, X. Munoz, D. Raba, J. Marti, and X. Cufi, ―Yet another survey on image segmentation: Region and boundary information integration,‖ in *Proc. 7th Eur. Conf. Computer Vision Part III*, Copenhagen,Denmark, May 2002, pp. 408–422, LNCS.
[23] A. Y. Yang, J. Wright, S. Sastry, and Y. Ma, ―Unsupervised segmentation of natural images via lossy data compression,‖ *Comput. Vis. Image Understand.*, 2007,\ submitted for publication.
[24] Y. Ma, H. Derksen, W. Hong, and J. Wright, ―Segmentation of multivariate mixed data via lossy coding and compression‖ *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1546–1562, Sep. 2007.
[25] A. Y. Yang, J. Wright, S. Sastry, and Y. Ma, ―Unsupervised segmentation of natural images via lossy data compression,‖ Elect. Eng. Comput. Sci. Dept. Univ. California, Berkeley, 2006 [Online].Available:http://www.eecs.berkeley.edu/Pubs/Tec hRpts/2006/EECS-2006-195.html

Article received: 2011-07-18