K-Means Cluster Analysis for Image Segmentation

S. M. Aqil Burney
Institute of Business Management
Karachi, CO 75270 Pakistan

Humera Tariq
University of Karachi
Karachi, CO 75270 Pakistan

ABSTRACT

Does K-Means reasonably divides the data into k groups is an important question that arises when one works on Image Segmentation? Which color space one should choose and how to ascertain that the k we determine is valid? The purpose of this study was to explore the answers to aforementioned questions. We perform K-Means on a number of 2-cluster, 3-cluster and k-cluster color images (k>3) in RGB and L*a*b* feature space. Ground truth (GT) images have been used to accomplish validation task. Silhouette analysis supports the peaks for given k-cluster image. Model accuracy in RGB space falls between 30% and 55% while in L*a*b* color space it ranges from 30% to 65%. Though few images used, but experimentation proves that K-Means significantly segment images much better in L*a*b* color space as compared to RGB feature space.

Keywords

Cluster evaluation, L*a*b* Color Space, Precision Recall Graph, Image Segmentation

1. INTRODUCTION

IMAGE SEGMENTATION is a process that splits a given Image 'I' into n sub regions. Robust literature on image segmentation is available [1-7] which states that for a given image I, Image segmentation process makes 'n' disjoint partitions in the image I. Let the partitions are represented by R_i ($i = 1,2,3............n$)then they must satisfy following properties:

   a) $\cup_{i=1}^{n} R_i = I$
   b) $R_i \cap R_j = \varphi$ $with\ i \neq j$
   c) $H(R_i) = TRUE$ $\forall i$
   d) $H(R_i \cup R_j) = FALSE$
       $\forall\ R_i$ and R_j adjacent

The first two properties show that the regions form a partition of Image 'I' since they split 'I' into disjoint subsets. The third property ensures that pixels with in particular cluster must share same featured components and finally if pixels belong to two different regions their feature components must also be different from each other. A very important aspect of segmentation process is H(R), the homogeneity attributes of pixels over region R and on the basis of which the whole segmentation process has been carried on. The homogeneity or similarity is measured by comparing the intensity, color and/or texture of different pixels and if they satisfy certain predefined criteria, they are supposed to belong to region R.

Primarily K-Means performs Identification of new and unknown classes according to given data set and segments the instance space into regions of similar objects. It is an unsupervised iterative and heuristic partitioning based clustering algorithm, first introduced by James MacQueen[8],[9]. The robust flavour of K-Means make it inevitable for unsupervised learning of neural network, Pattern recognitions, Classification analysis, Artificial intelligent, image processing and machine vision. K-Means makes hard assignments of points to clusters in contrast to EM for Gaussian Mixture Model GMM [10]-[13] and Fuzzy K-Means [14]-[16] which allow for soft assignments. Estimating centroids from data in K-Means is computationally expensive and not guaranteed to produce satisfactory result. It has been noted that K-Means result often suffers from local minima, noise and over segmentation.

Color image segmentation uses color as homogeneity criteria for grouping. Varieties of K-Means descendent for color image segmentation have been explored by research scholars over the decades. The non- linear transformation of RGB color components to device independent CIE-L*a*b* space and CIE-L*u*v* has been opted frequently in the past [17],[18] and recent years [19],[20].The reason of being doing so is to establish an isotropic feature space. Both color spaces are uniform derivations from the standard CIE-XYZ space. In L*u*v Color space the chromatic information is carried by u* and v* while the lightness coordinate L* represents the relative brightness. The three coordinates of CIE-L*a*b* are L*, a* and b* where a* denotes the red/green value and b* the yellow/blue value. L.Lucchese and S.K.Mitra [21] proposed a novel algorithm based on K-Means clustering where they apply 2D K-Means in chromaticity plane and combine its results with intensity based 1-d k- mean algorithm. Soon Chi Zhang and P.Wang [22] presents their work on color image segmentation in HSI space where H represents Hue, S represents saturation while I denotes intensity. They applied K-Means on medical images and showed that hue image clustering results are quite promising. HSV color space has also been worked out, where Wei, Ling and Shao [23] successfully address three main issues with K-Means. They reduce the computational complexity of K-Means by using histogram quantization in HSV color space. They estimate the number of cluster of K-Means without testing each value of k. Their algorithm also addressed over- segmentation problem of K-Means. Their segmentation results were close to human perception. The hybrid K-Means approach proposed in [24] incorporates spatial coordinates with K-Means in a novel way. They didn't use spatial information in K-Means feature space rather they employ it for carrying region growing and the outcome of this process was passed to K-Means as a feature vector for further refinements.
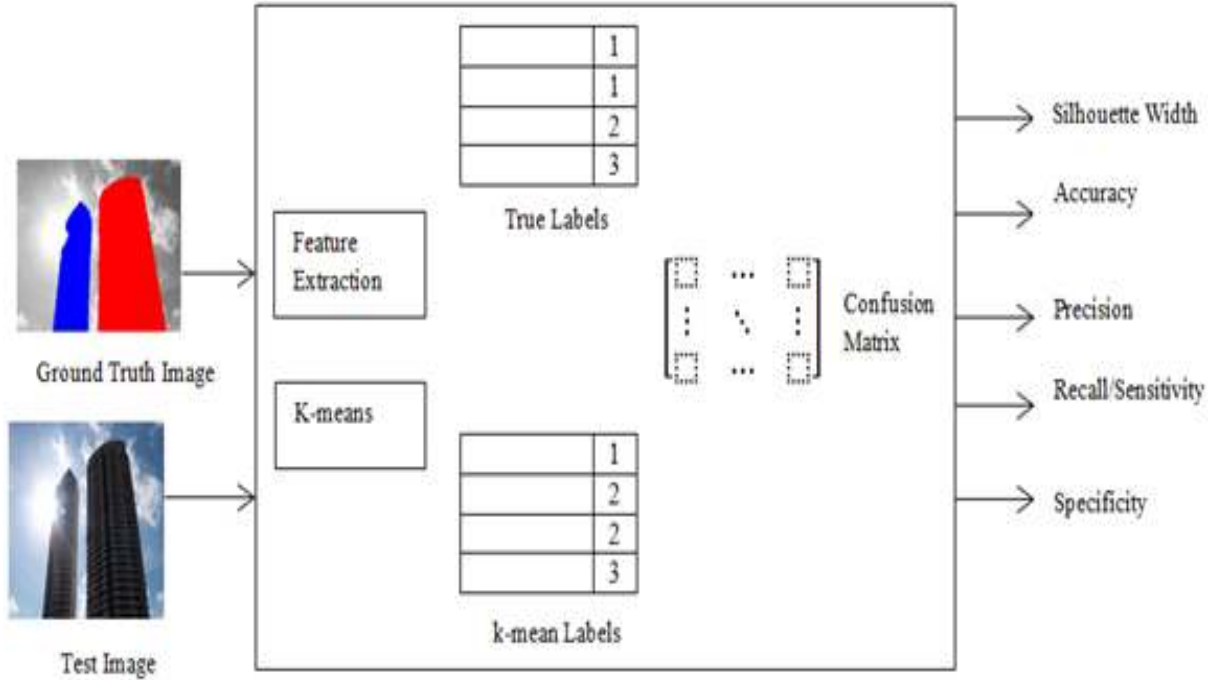
Fig 1: Ground Truth Based K-Means Evaluation System

As a result the four dimensional feature spaces were composed of three color components and texture as a fourth one. So K-Means is playing full fledge role for color image segmentation in a versatile and fruitful way since its birth [25].

This paper provides an overview of K-Means clustering method for color image segmentation along with custom employment of color space that has been appeared in the past and recent literature. A full description of these methods is beyond the scope and the readers are referred to references for details. We implement and test the performance of K-Means in RGB and L*a*b* color space by comparing the K-Means labeling with manually labeled ground truth images. We did a strong analysis based on silhouette width of pixels and more appropriately by computing confusion matrix base metrics. The metrics measures the accuracy, precision, sensitivity and specificity of clusters and then compute macro averages to yield a point on Precision Recall (PR) space.

Fig. 1 demonstrates the schematic of the system to evaluate the performance of K-Means via ground truth. The Ground truth labels are compared with K-Means labels to build confusion matrix on the basis of which cluster evaluation parameters will be determined that validates the K-Means clustering.

The rest of the paper is organized as follows: Section II details each step of the K-Means algorithm. Cluster evaluation and confusion based metrics were discussed in Section III. Section IV presents our Experiment details and results. Finally conclusion is presented in Section V.

2. K-MEANS ALGORITHM

We assume the number of clusters, k, is given. We use the center of each cluster C_i to represent each cluster. The center of each cluster is the mean of the data points which belong to such a cluster. Basically, we define a distance measurement $D(x, y) = \|x - y\|^2$ to determine which cluster a data point belongs to? We can compare the distance of a data point to

these cluster centers and a data point x_k belongs to its nearest cluster:

$$l_k(x_k) = arg \min_i D(x_k - C_i) = arg \min_i \|x_k - C_i\|^2$$

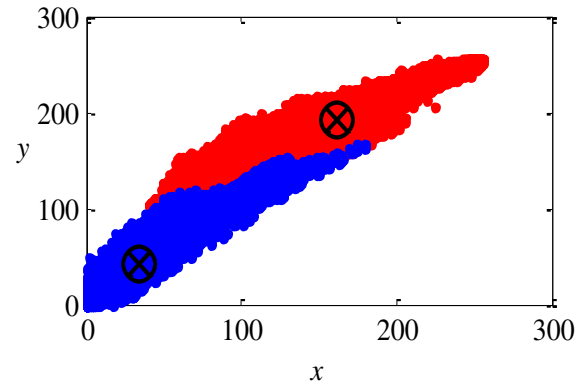Where l_k is the label for the data point x_k .

Fig 2: Basic K-Means Algorithm

The K-Means algorithm tries to find a set of such cluster centers such that the total distortion is minimized. Here the distortion is defined by the total summation of the distances of data points from its cluster center:

$$\emptyset(X, C) = \sum_{i \in C} \sum_{j \in ith\ cluster} \|x_j - C_i\|^2$$

To minimize $\emptyset$, K-Means algorithm iterates between two steps: Labeling and Re-centering.

Labeling: Assume the p–th iteration ends up with a set of cluster centers $C_i^{(p)}$, $i = 1, \dots \dots, K$. We label each data point based on such a set of cluster centers, i.e., $\forall\ x_k$, find

$$C_i^{(p+1)} = \frac{\sum_{x_k \in \Omega_i} x_k}{|\Omega_i|}$$

$$l_k^{(p+1)}(x_k) = \min_i \|x_k - C_i^p\|^2$$

And group data points belong to the same cluster

$$\Omega_j = \{x_k : l_k(x_k) = C_j\}$$

Re-Centering: Recalculating the centers for all the clusters

The algorithm iterates between such labeling and re-centering steps and is expected to converge to a local stationary status.

In pseudo code, K-Means is shown by [26] to follow this procedure:

> Initialize $\mathbf{m}_i$, $i = 1,…,k$, for example, to k random $\mathbf{x}^t$
> Repeat
>
> For all $\mathbf{x}^t$ in X
> $b_i^t \leftarrow 1$ if $\| \mathbf{x}^t - \mathbf{m}_i \| = \min_j \| \mathbf{x}^t - \mathbf{m}_j \|$
> $b_i^t \leftarrow 0$ otherwise
> For all $\mathbf{m}_i$, $i = 1,…,k$
> $\mathbf{m}_i \leftarrow$ sum over t $(b_i^t \mathbf{x}^t)$ / sum over t (b_i^t)
>
> Until $\mathbf{m}_i$ converge

The vector $\mathbf{m}$ contains a reference to the sample mean of each cluster. $\mathbf{x}$ refers to each of our examples, and $\mathbf{b}$ contains our "estimated class labels".

3. CLUSTER EVALUATION METRICS

Clustering algorithms can be hard to evaluate *without prior information* or assumptions on the data. With no assumptions on the data, one evaluation technique is w.r.t some ground truth image. The ground truth images are not unique and thus more than one segmentation of given image needs to consider. The problem become more cumbersome when the clusters are supposed to show more than two or three distinguished objects with in a given image. The brief discussion of the metrics that we use to test our experimentation results have been discussed in this section.

3.1 Silhouette Width

Each cluster is represented by a so-called silhouette, which is based on the comparison of its tightness and separation. This silhouette shows which objects were placed well within their cluster, and which ones are merely somewhere in between clusters. The entire clustering is displayed by combining the silhouettes into a single plot, allowing an appreciation of the relative quality of the clusters and an overview of the data configuration. The average silhouette width is an optimal ratio that maximizes inter-clusters distance and minimizes intra-clusters distance. This mean silhouette provides an evaluation of clustering validity, and might be used to select an 'appropriate' number of clusters K.

For each cluster *i*, the silhouette width *s(i)* is defined as follows: Let *a(i)* be the average dissimilarity between *ith* pixel and all other pixels in its cluster. For all other clusters *C* to which *ith* pixel is not assigned, let $d(i, C)$ be the average dissimilarity of *ith* pixel to all of the pixels in *C*. The smallest of these distances $d(i, C)$ is represented by $b(i)$, where $b(i) = \min_C d(i, C)$. In words, $b(i)$) is the dissimilarity between *ith* pixel and the nearest cluster to which it doe*s not* belong. This means to say that *s(i)* is :

$$s(i) = \frac{b(i) - a(i)}{\max (a(i), b(i))}$$

Observations with a large *s(i)* (almost 1) are very well clustered, a small *s(i)* (around 0) means that the observation lies between two clusters, and observations with a negative *s(i)* are probably placed in the wrong cluster[27].

Table 1
Interpretation of Silhouette Values

Mean Silhouette	Interpretation
0.71- 1.00	Excellent Split
0.51 – 0.70	Reasonable Split
0.26 – 0.50	Weak Split
≤ 0.25	Bad Split

3.2 Confusion Matrix

A confusion matrix [28] contains information about actual and predicted classifications done by a classification system. Performance of such systems is commonly evaluated using the data in the matrix. Table II shows the confusion matrix for a two class classifier. The entries in the confusion matrix have the following meaning in the context of our study: True Positive (TP) is the number of correct predication that a pixel belongs to a particular cluster, True Negative (TN) is the number of correct prediction that a pixel does not belong to a particular cluster, False Positive (FP) is the number of incorrect predictions that a pixel belongs to cluster, and FN is the number of incorrect predictions that a pixel does not belong to a cluster.

Table II
Confusion Matrix

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP (type I error)
	Negative	FN (type II error)	TN

Several standard terms have been defined for the 2 class matrix: The accuracy/precision (AC) is the proportion of the total number of predictions that were correct. Therefore Accuracy is determined using the equation:

$$AC = \frac{TP + TN}{TP + TN + FP + FN}$$

Though Accuracy is the most basic and ubiquitous metric but it has some serious limitations.

Precision is a measure whose denominator involves only the positive predictions by classification system. It is the percentage of the number of *true positive pixels* divided by the

sum of the number of *true positive pixels* and false positive *pixels*.

$$Precision = \frac{TP}{TP + FP}$$

The recall or sensitivity represents the true positive rate of the class. It is the proportion of positive cases that were correctly identified, as calculated using the equation:

$$recall/sensitivity = \frac{TP}{TP + FN}$$

Typically, there's a trade-off between precision and recall.

Another important measure is specificity which reflects true negative rate of class. It is usually used in binary classification.

$$Specificity = \frac{TN}{TN + FP}$$

3.3 Precision Recall Graph

In image segmentation the labels to be deal with are discrete so K-Means produces a (Sensitivity, Precision) pair, which corresponds to a single point on Precision/Recall (P/R) graph [29], [30]. A basic P/R graph has Recall on X-axis and Precision on Y-axis. Fig. 3 shows a P/R graph for discrete classifier. There exist a trade-off between precision and recall. A high precision means low recall. If the point on P/R graph lies on the diagonal line between (0, 0) and (1, 1) it means the classifier randomly predict a class i.e. it expect fifty percent positive and fifty percent negative. A classifier with point on the Top Left portion of the graph is considered to be more precise (true positive) and least sensitive (false positive). The performance of classifier is considered poor if a point on ROC graph is located on Bottom Right.
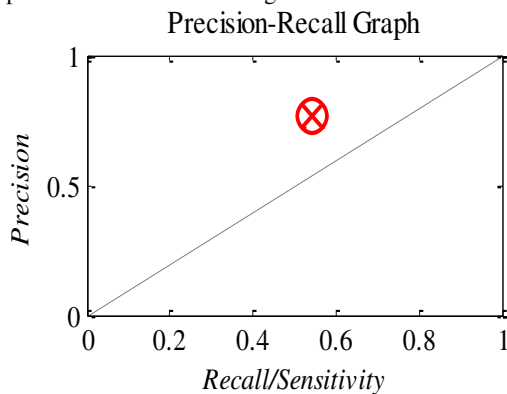

Fig 3: Precision /Recall Graph

4. EXPERIMENTATION

4.1 Dataset

To segment 2-cluster and 3-cluster images we have used images from Alpert et al. [35], and *Weizmann horse database* [31-33]. These databases contain salient segmentation of images. For k-cluster input where k>3 we use *Berkeley segmentation database* [34]. The BSD provides 300 images and silhouette segmentations by up to 300 subjects. Example images along with their ground truth are shown in Fig. 3. These images are referred throughout the paper as tree.png, pigeon.png, horse.png and boy.png.
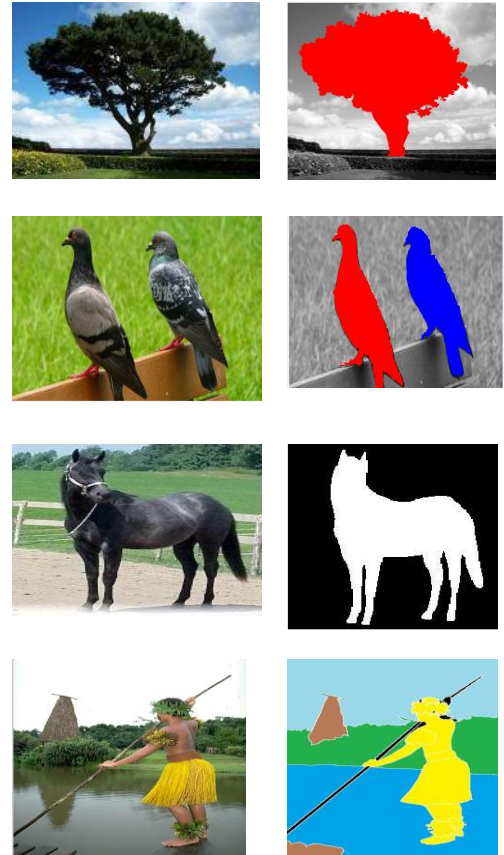

Fig 4: Images tree.png, pigeon.png, horse.png, boy.png along with their Ground Truth

4.2 Features and K-Means labeling

Each pixel in an image has certain features according to which it could be clustered with others, leading to a segment. We select the pixel's spatial coordinates and L*a*b* components to be its set of distinguishing feature and create 5 dimensional feature vector from it. Let the classes in a particular image G be $C = (c_1, c_2, ..., c_k)$. The clustering method produces k clusters, which divides G into k disjoint subsets, R_1, R_2, ..., R_k. We assume that each class is a cluster and apply K-Means clustering which assign each pixel to its nearest centroid so that the Euclidean distance between them is minimized in the mentioned 5d feature space. K-Means fairly partitions the input image into significant regions though the results becomes vary if we change k and re-run it. It can also be seen that if k is increased more partitions will be added and more intensities are available.

Table III Time to Compute Mean Silhouette Values		
K	Mean Silhouette Value	Time (Seconds)
2	0.8506	365
3	0.7877	375
4	0.7554	398
5	0.7383	408
6	0.7195	410
7	0.6962	403

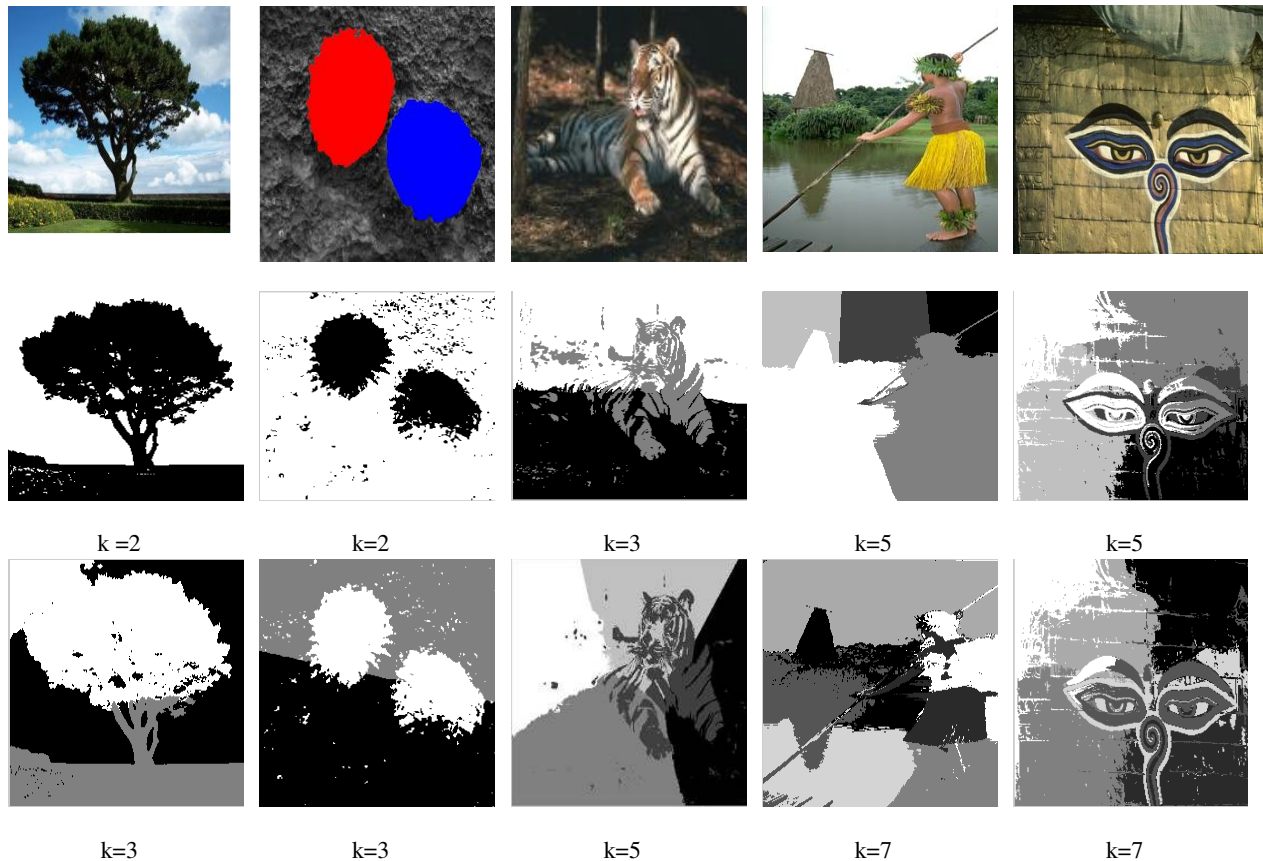| k =2 | k=2 | k=3 | k=5 | k=5 |

| k=3 | k=3 | k=5 | k=7 | k=7 |

Fig 5:   K-Means results for 2-cluster, 3-cluster and k-cluster Images (k>3)

4.3 Silhouette Analysis

For cluster validation we run K-Means a bunch of times, each time with a different *k*. Fig. 6 (a) shows the silhouette width per cluster. From silhouette plot for 'boy.png' in Fig. 6 it can be seen that most points in all clusters have a large silhouette value, greater than 0.6, indicating that those points are well-separated from neighboring clusters. Fig 6 (b) shows graph of mean silhouette value versus k with same k= 5 for 'boy.png' Peak in mean silhouette value was found at k= 4 and k=7 for 'boy.png', which seem reasonable. In all our experimental results the K-Means attain its first peak at k =2 indicating that K-Means identifies clusters in its early iterations. As the value of k becomes greater than 2 the K-Means algorithm result varies in drastic manner. A noticeable point in silhouette analysis is that computation of this metric is extremely time consuming. Table III shows the time consumed in seconds to compute the mean silhouette value for 'boy.png'
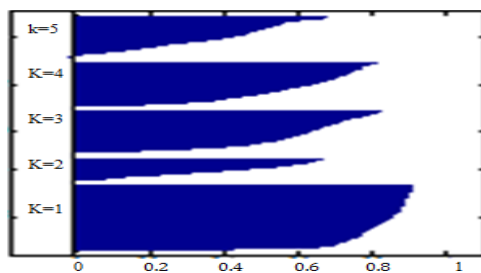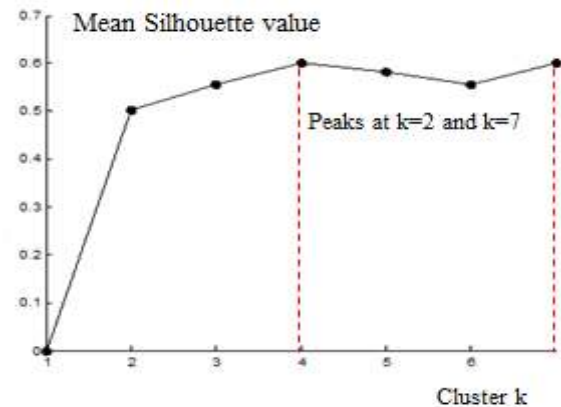

Fig 6 (b): Cluster k vs. Mean Silhouette Value for 'boy.png'

K-Means with different k is run for many images but here we include result only for 'boy.png' and 'tiger.png' with k =5. So Fig 6(c) and Fig 6(d) represents silhouette plot and mean silhouette value for tiger.png respectively. This time silhouette value exceeds 0.8 for all clusters showing good separation in feature space. But the graph of Mean silhouette suggests three values of k which are 3, 4 and 5, hence depending on how many clusters the user is interested, an appropriate k can be chosen. Some negative silhouette values are also observed showing the similarity of pixels with other centroid or cluster as well.
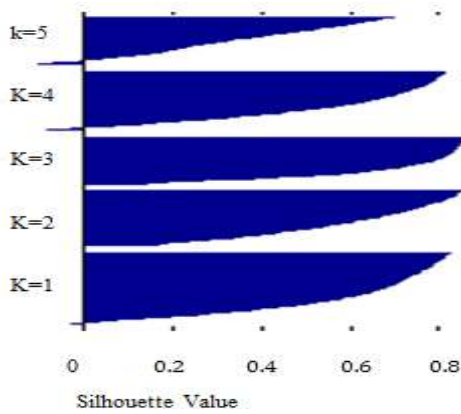

Fig 6 (a): Silhouette width vs. cluster k for ' boy.png'

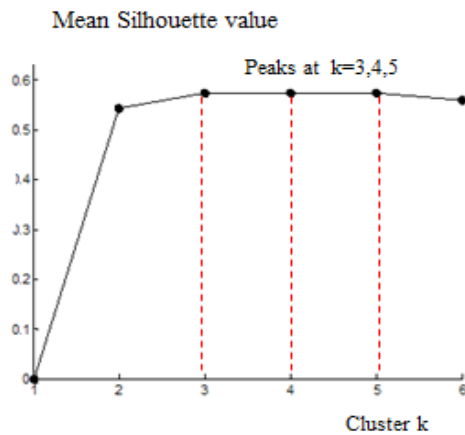**Fig 6 (c):Silhouette width vs. cluster k for 'tiger.png'**

Fig 6 (d): Cluster k vs. Mean Silhouette Value for 'tiger.png'

4.4 Confusion Matrix Analysis

To Judge the clustering quantitatively confusion matrix has been built using ground truth labeling and the labeling predicted by K-Means in RGB and L*a*b* color spaces. Confusion Matrices for 2-cluster Image 'tree.png' in RGB, and L*a*b* color space are shown in Table IV and Table V respectively. It can be seen in Table IV and Table V that for 'tree.png' K-Means TP lies between 50% to 63%; FN lies between 32% to 45% showing K-Means power as classifier. Table VI which contains statistics for 'horse.png' also reflects that K-Means has high sensitivity and high specificity confirming K-Means strength as classifier. It can also be seen here that K-Means show least FP and least TN in this case. With binary image of horse the classifier guess is entirely random. So K-Means performs badly when there is only black and white to discriminate it from the rest of the images.(See Tables VI and VII for 'horse.png'). The confusion matrix for 'pigeon.png' (k=3) is shown in Table VIII and Table IX. This time K-Means performs poorly in both color spaces which adhere the need of incorporating more features for discrimination amongst pixels in feature space.

Table IV
RGB Confusion Matrix ' Tree.png'

GT K-Means	background	Not Background
background	50.5%	0.1%
Not background	45.2%	4.2%

Table V
L*a*b* Confusion Matrix'tree.png'

GT K-Means	background	Tree
background	63.1%	1.1%
tree	32.6%	3.2%

Table VI
RGB Confusion Matrix 'Horse.png'

GT K-Means	background	horse
background	29.2%	26.1
horse	21.3%	23.4%

Table VII
L*a*b* Confusion Matrix 'Horse.png'

GT K-Means	background	horse
background	24.0%	2.8%
horse	26.7%	46.7%

Table VIII
L*a*b* Confusion Matrix 'Pigeon.png'

GT K-Means	background	Red Pigeon	Blue Pigeon
background	40%	9%	13%
Red Pigeon	0.3%	12%	13%
Blue Pigeon	4%	5%	4%

Table IX
L*a*b* Confusion Matrix 'Pigeon.png'

GT K-Means	background	Red Pigeon	Blue Pigeon
background	40%	9%	13%
Red Pigeon	0.3%	12%	13%
Blue Pigeon	4%	5%	4%

All experimental results of accuracy, precision, sensitivity and specificity are tabulated in Table X. The performance via Recall/Sensitivity graph of K-Means in RGB space is shown in Fig. 7 while the Recall/Sensitivity graph of K-Means performance in L*a*b* space, is shown in Fig. 8 along with corresponding feature space.

Table X
Summary of Result for Images used in Experiments

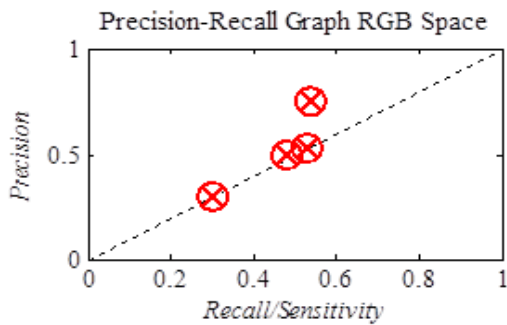|	Feature Space	Accuracy	Precision	Sensitivity	Specificity
Tree K=2	L*a*b*	0.66	0.54	0.71	0.71
	RGB	0.55	0.54	0.75	0.75
Horse K=2	L*a*b*	0.71	0.77	0.71	0.71
	RGB	0.53	0.53	0.53	0.53
Pigeon K=3	L*a*b*	0.57	0.48	0.50	0.77
	RGB	0.56	0.50	0.50	0.76
Boy K=6	L*a*b*	0.46	0.30	0.33	0.89
	RGB	0.39	0.30	0.30	0.88

Fig 7: Performance of K-Means in RGB Space

5. CONCLUSION AND FUTURE WORK

K-Means is an algorithm that quickly group pixels on the basis of predefined feature vectors and initial centroids. The consideration of color components alone as feature space dimension is not enough to validate the results for clustering and hence an improvement by means of parameterization, feature space extension with texture or gradient is necessary. The chosen color spaces have also shown a substantial effect on clustering results and accuracy. We found L*a*b* space is better in comparison with RGB space for clustering in precision recall space. Silhouette analysis is a useful criterion to asses a candidate k for clustering. Finally confusion matrix allows one to visualize and measure the performance of K-Means algorithm but it needs ground truth to evaluate the performance. In future other K-Means extensions will be assessed for comparison with more type of images for effective validation of experimental results.
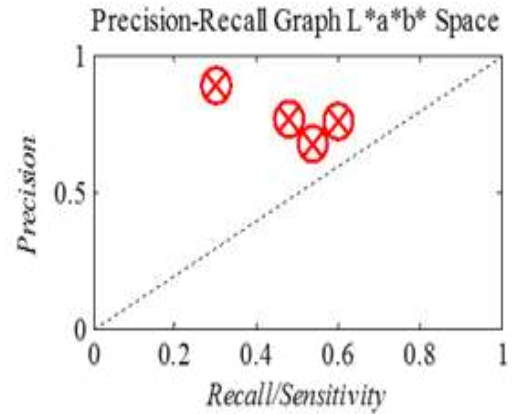

Fig 8: Performance of K-Means in L*a*b* Space

6. REFERENCES

[1] L.Lucchese, S.K. Mitra, "Color Image Segmentation: A State of the Art Survey," (invited paper), Image Processing, Vision and Pattern Recognition, Proc. of Indian National Science Academy, vol. 67, A, No.2, pp. 207-221, March-2001.

[2] H. D. Cheng , X. H. Jiang , Y. Sun , Jing Li Wang, "Color image segmentation: Advances and prospects (2001)",Pattern Recognition, 34, 2001, 2259 - 2281

[3] W.Skarbek, A.Koschan, "Colour Image Segmentation- A Survey", Technical Report 94-32, Technical University Berlin, October. 1994

[4] K.S.Fu, J.K.Mui,"A Survey on Image Segmentation," Pattern Recognition, Vol. 14, pp.3-16, 1981.

[5] N.R.Pal and S.K.Pal, "A Review on Image Segmentation Techniques," Pattern Recognition, Vol26, No. 9, pp.1277-1294, 1993.

[6] Haralick, R. M. and Shapiro, L. G., Survey: "Image Segmentation Techniques". CVGIP, vol. 29, 1985,pp. 100-132.

[7] G. B. Coleman and H. C. Andrews "Image segmentation by clustering", Proc. IEEE, vol. 67, pp.773 -785 1979

[8] Lloyd SP "Least squares quantization in PCM. Unpublished Bell Lab. Tech. Note, portions presented at the Institute of Mathematical statistics Meeting Atlantic City, NJ, sep. 1957.

[9] MacQueen, J. (1967) "Some methods for classification and analysis of multivariate observations". In Proc. 5th Symp. Math Statist Prob (pp. 281–297).

[10] Arthur Dempster, Nan Laird, and Donald Rubin. "Maximum likelihood from incomplete data via the EM algorithm". Journal of the Royal Statistical Society, Series B, 39(1):1–38, 1977

[11] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, New York, 1996

[12] G. McLachlan and D. Peel, Finite Mixture Models, John Wiley & Sons, Brisbane, 2001.

[13] P.Haim, F.Joseph, and J. Ian, "A study of Gaussian mixture models for color and texture features for image classification and segmentation," Pattern Recognition, vol 39, no.4, pp.695-706, 2006.

[14] J.K. Udupa, S. Samarasekera, "Fuzzy connectedness and object definition: theory, algorithm and applications in image segmentation, Graph. Models Image Process", 58(3) (1996) 246–261.

[15] S.M. Yamany, A.A. Farag and S. Hsu, "A fuzzy hyperspectral classifier for automatic target recognition (ATR) systems", Pattern Recognit. Lett., 20 (1999)1431–1438

[16] G. C. Karmakar and L. S. Dooley, "A generic fuzzy rule based image segmentation algorithm", Pattern Recognition Letters, 23(10) (2002) 1215-1227

[17] W. Woelker, "Image Segmentation Based on an Adaptive 3D Analysis of the CIE-L*a*b* Color Space," Proc. of SPIE'96- Visual communications and Image Processing '96, Vol.2727, pp. 1197-1203, 1996

[18] D. Comaniciu, P. Meer, "Robust Analysis of Feature Spaces:Color ImageSegmentation," Proc.of CVPR'97, pp. 750-755

[19] H.D. Cheng and Y. Sun, "A hierarchical approach to color image segmentation using homogeneity", IEEE Transactions on Image Processing 9 (12) (2000)2071–2082

[20] Yonghong Zhang , "EM algorithm with improvement for color image segmentation in multiple color spaces", Electronics, Communications and Control (ICECC), 2011, pp.853 - 857

[21] L. Lucchese and S. K. Mitra, "Unsupervised segmentation of color images based on K-Means clustering in the chromaticity plane", Proc. of Content-based access of image and video libraries, pp. 74-78, 1999.

[22] Chi zhang, P. Wang, A New Method of Color Image Segmentation Based on Intensity and Hue Clustering, volume 3 2002.

[23] Tse-Wei Chen , Yi-Ling Chen and Shao-Yi Chien (2010), "Fast Image Segmentation Based on K-Means Clustering with Histograms in HSV Color Space," Journal of Scientific Research ISSN I452-2I6X Vol. 44 No.2, pp.337-35 I

[24] F. Gibou and R. Fedkiw, "A fast hybrid K-Means level set algorithm for segmentation," in Proc. 4th Annu. Hawaii Int. Conf. Stat. Math., 2005,pp. 281–291.

[25] De Montréal, "A De-Texturing and Spatially Constrained K-Means Approach for Image Segmentation Max Mignotte", ELSEVIER, Pattern Recognition Letters 32 (2011) pp. 359-367.

[26] Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, 2004.

[27] L. Kaufman and P. J. Rousseeuw. Finding groups in data: an introduction to cluster analysis. John Wiley and Sons, New York, 1990.

[28] Provost Foster J., Fawcett Tom, and Kohavi Ron, "The case against accuracy estimation for comparing induction algorithms". Proceedings of the Fifteenth International Conference on Machine Learning, Morgan Kaufmann Publishers Inc , pages 445–453,1998.

[29] T. Fawcett, "ROC Graphs: Notes and Practical Considerations for Researchers",Intelligent Enterprise Technologies Laboratory, 2004.

[30] M. Zweig and G. Campbell, "Receiver-Operating Characteristic (ROC) Plots: A Fundamental Evaluation Tool in Clinical Medicine", Clin. Chem. 39/4, 561-577, 1993.

[31] Borenstein and J. Malik. Shape guided object segmentation. In IEEE Conference on Computer Vision and Pattern Recognition, 2006.

[32] E. Borenstein, E. Sharon, and S. Ullman. Combining top-down and bottom-up segmentation. In Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04).

[33] E. Borenstein and S. Ullman. Class-specific, top- down segmentation. In European Conference on Computer Vision, 2002.

[34] 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," Proc. IEEE Eighth Int'l Conf. Computer Vision, pp. 416-423, July 2001.

[35] http://www.wisdom.weizmann.ac.il/~vision/Seg_Evaluation_DB/index.html