CTEX ALGORITHM FOR IMAGE SEGMENTATION

Shital A. Kulkarni[*]

ABSTRACT:

Segmentation refers to the process of partitioning a digital image into sets of pixels or superpixels. The goal of segmentation is to simplify and change the representation of an image for meaningful analysis. Image segmentation has wide applications as Medical Imaging for diagnosis, treatment planning, computer aided surgery, face recognition, fingerprint recognition, traffic control systems, brake light detection, machine vision etc. In this paper, natural images are segmented by combing color and texture information. For this an unsupervised image segmentation framework (referred as CTex) is used. CTex is based on the adaptive inclusion of color and texture in the process of data partition. It is new formulation for the extraction of color features that is evaluated using the input image in a multispace color representation. The key component is the inclusion of the Self Organizing Map (SOM) network in the computation of the dominant colors and estimation of the optimal number of clusters in the image. The texture features are computed using a multichannel texture decomposition scheme based on Gabor filtering. This paper is implemented using the image processing, neural network and the statistical toolboxes of Matlab.

Key words: *Image Segmentation, SOM, CTex, Gabor filtering, image processing, neural network and statistical toolboxes.*

[*] M. E. Electronics, Assistant Professor, E&TC Department, PICT, Pune, Maharashtra

INTRODUCTION:

Segmentation is a process of grouping an image into units that are homogeneous with respect to one or more characteristics. In past decades, attention has been focused on monochrome image segmentation whose goal is to separate individual objects in the perception of the scene. A common problem in segmentation of monochrome image occurs when an image has a background of varying gray levels such as gradually changing shades. The early color-texture segmentation algorithms were designed in conjunction with particular applications and they were generally restricted to the segmentation of images that are composed of scenes defined by regions with uniform characteristics. Segmentation of natural images is by far a more difficult task, since natural images exhibit significant inhomogeneities in color and texture. In fact, the textures that are present in natural images are often characterized by a high degree of complexity, randomness, and irregularity.

COLOR SEGMENTATION ALGORITHM:

GB-FAB ANISOTROPIC FILTERING

The adaptive filtering technique that has been implemented for preprocessing the input image is an improvement of the forward and backward anisotropic diffusion (also called FAB). The FAB anisotropic diffusion is a nonlinear feature preserving smoothing technique that efficiently eliminates the image noise and weak textures from the image while preserving the edge information.

The original anisotropic diffusion filtering has been proposed by Perona and Malik where they formulated the smoothing as a diffusive process that is performed within the image regions and suppressed at the regions boundaries. In order to achieve this behavior, a mathematical framework where the central part is played by a diffusion function that controls the level of smoothing is developed.

$$\frac{\partial \mathrm{I}(x,y,t)}{\partial t} = div[D(|\nabla \mathrm{I}(x,y,t)|)\nabla \mathrm{I}(x,y,t)] \text{ ....Eqn. (1)}$$

Where I(x, y) is the image data, ∇ I (x, y, t) is the gradient operator at the position (x, y) at iteration t, D (˙) represents the diffusion function and *div* is the divergence operator. The function D is usually implemented using an exponential function as illustrated in Eqn. (2), where the parameter controls the level of smoothing

$$D(|\nabla \mathrm{I}(x,y,t)|) = e - (|\nabla \mathrm{I}(x,y,t)|/d)^2, d > 0 \text{ ....Eqn. (2)}$$

It can be observed that the diffusion function D (·) is bounded in the interval (0, 1) and decays with the increase of the gradient value ∇I. To eliminate the limitations associated with the original PM formulation, the FAB anisotropic diffusion has been proposed. The goal of the FAB diffusion function is to highlight the medium and large gradients that are noise independent and this is achieved by reversing the diffusion process. This can be implemented by applying two diffusions simultaneously: the forward diffusion that acts upon the low gradients that are usually caused by noise, while the backward diffusion is applied to reverse the diffusion process when dealing with medium gradients. Nonetheless, since the D_{FAB} function is defined by two parameters the problems associated with stability are more difficult to control. To address these problems, Smolka and Plataniot is proposed the inclusion of a time dependent cooling procedure where the values of the diffusion parameters are progressively reduced with the increase in the number of iterations

$$D_{FAB}(|\nabla \mathrm{I}(x,y,t)|) = 2e - (|\nabla \mathrm{I}(x,y,t)|/d1(t))^2 - e - (|\nabla \mathrm{I}(x,y,t)|/d2(t))^2 \text{ ....Eqn. (3)}$$

$$\begin{array}{l} di(t+1) = di(t).\gamma, i = 1,2 \\ di(t+1) < di(t), \gamma \in (0,1) \end{array} \text{ ....Eqn. (4)}$$

Where d_1 (t=0) is the starting parameter, ∇I represents the image gradient, γ is a fixed parameter that takes values in the interval (0, 1), d_1(t) and d_2(t) are the time dependent parameters that control the forward and backward diffusion respectively and t is the time or iteration step. In this implementation these parameters are set to the following default values: d_1 (t=0) = 40, d_2 (t=0) = 2 d_1 (t=0) = 80 and γ=0.8.

To further reduce the level of blurriness, a boosting function is proposed to amplify the medium gradients and obtain much crisper image details. The gradient value will be replaced by the new “boosted” value.

$$|\nabla \mathrm{I}(x,y,t)| \leftarrow |\nabla \mathrm{I}(x,y,t)|(1 + 2e - \|\nabla \mathrm{I}(x,y,t)\| - m/d1(t) \text{ ....Eqn. (5)}$$

Where m is the median value of the gradient data. Significantly improved results are obtained when the input image is filtered with the GB-FAB algorithm.

DOMINANT COLOR EXTRACTION:

The performance of the clustering algorithms is highly influenced by the selection of this parameter and an efficient solution to automatically detect the dominant colors and the final number of clusters in the image using a classification procedure based on the Self Organizing Maps (SOM) is proposed. Using the SOM, train a set of input vectors in order to obtain a lower dimensional representation of the input image in the form of a feature map that maintains the topological relationship and metric within the training set. The SOM networks were first introduced by Kohonen and they became popular due to their ability to learn the classification of a training set without any external supervision. In this implementation, a 2-D SOM network is created that is composed of nodes or cells [see Fig 1 (a)]. Each node Ni ($i \in [1, M]$) (where M is the number of nodes in the network) has assigned a 3-D weight vector that matches the size of each element of the input vector. It is important to mention that the training dataset represented by the input image is organized as a 1-D vector Vj (j=1…n, where n is the total number of pixels in the image) in a raster scan manner. Each element of the training set Vj is defined by a 3-D vector whose components are the normalized R, G, B values of the pixels in the image and is connected to all the cells of the network.[see Fig 1(a)].

In line with other clustering schemes, before starting the training procedure there is need to initialize the weights wi for all cells in the network. In practice, the random initialization is usually adopted when working with SOM networks and this is motivated by the fact that after several hundreds of iterations, the corresponding values of the initial random weights will change in accordance to the color content of the image. This procedure has been applied where the authors initialized the SOM network by randomly picking color samples from the input image. However, the random selection of the starting condition is sub-optimal since the algorithm can be initialized on outliers. Therefore, initialize the weights of the nodes in the SOM network with the dominant colors that are represented by the peaks (Pi) in the 3-D color histogram, calculated from the image that has been subjected to color quantization. This is achieved by applying a color quantization procedure that consists of re-sampling linearly the number of colors on each color axis. It has been experimentally demonstrated that a quantization value of 8 is sufficient to sample the statistical relevant peaks in the 3-D histogram. Thus, the quantized version of the input image is re-mapped so that the initial number of grey levels in all color bands 256×256× 256 is now reduced to 8×8×8. After constructing the 3-D histogram in the quantized color space, the peaks in relation to the desired number of dominant colors are selected by applying a quick sort algorithm. Considering that the size of the SOM lattice is four by four (i.e., M=16 cells), the first 16 highest histogram peaks are sufficient to accurately sample the dominant colors in the image.

AUTOMATIC DETECTION OF NUMBER OF CLUSTERS:

Once the initialization is completed ($wi \longleftarrow pi, i \in [1,16]$), the classification procedure is iteratively applied and consists of assigning the input vectors to the cell in the network whose corresponding weight values are most similar. The node in the SOM network that returns the smallest Euclidean distance is declared the best matching unit (BMU).

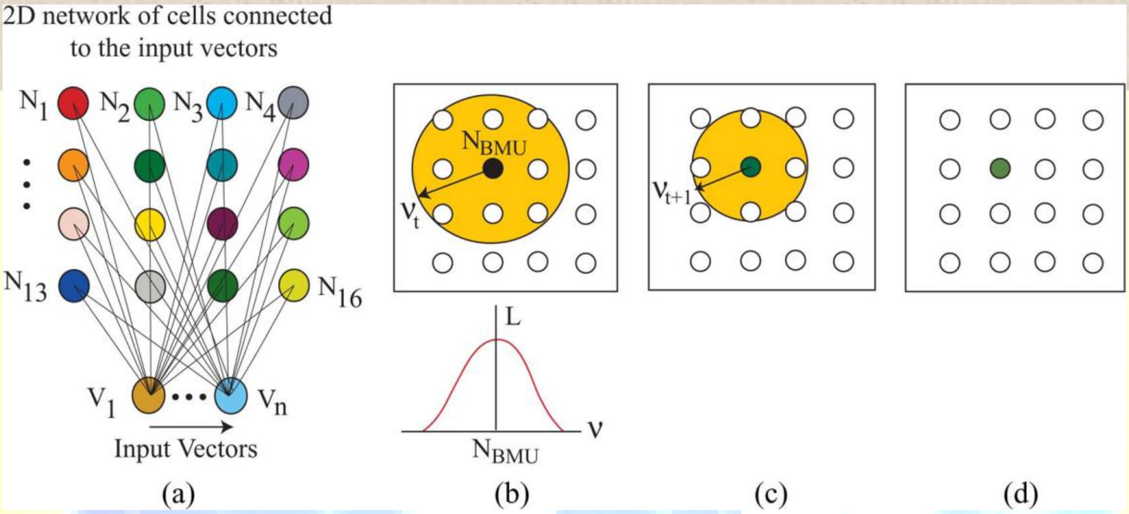


Fig 1 : (a) 2-D SOM network.
(b) The neighborhood of N at iteration t. The learning process of each cell's weight follows a Gaussian function, i.e., it is stronger for cells near node N and weaker for distant cells.
(c), (d) The radius $v(t)$ is progressively reduced until it reaches the size of one cell (N).

$$BMU = \arg\min \|Vj - Wi\| \quad i \in [1,16] \quad j \in [1,n] \qquad \text{...Eqn. (6)}$$

The weights of the N_{BMU} and of the nodes situated in its neighborhood are updated using the following learning rule:

$$wi(t+1) = wi(t) + L(t)[Vj(t) - wi(t)] \quad if \|N_{BMU} - Ni\| \le \nu(t) \quad wi(t+1) = wi(t), if \|N_{BMU} - Ni\| > \nu(t) \qquad \text{..................................Eqn. (7)}$$

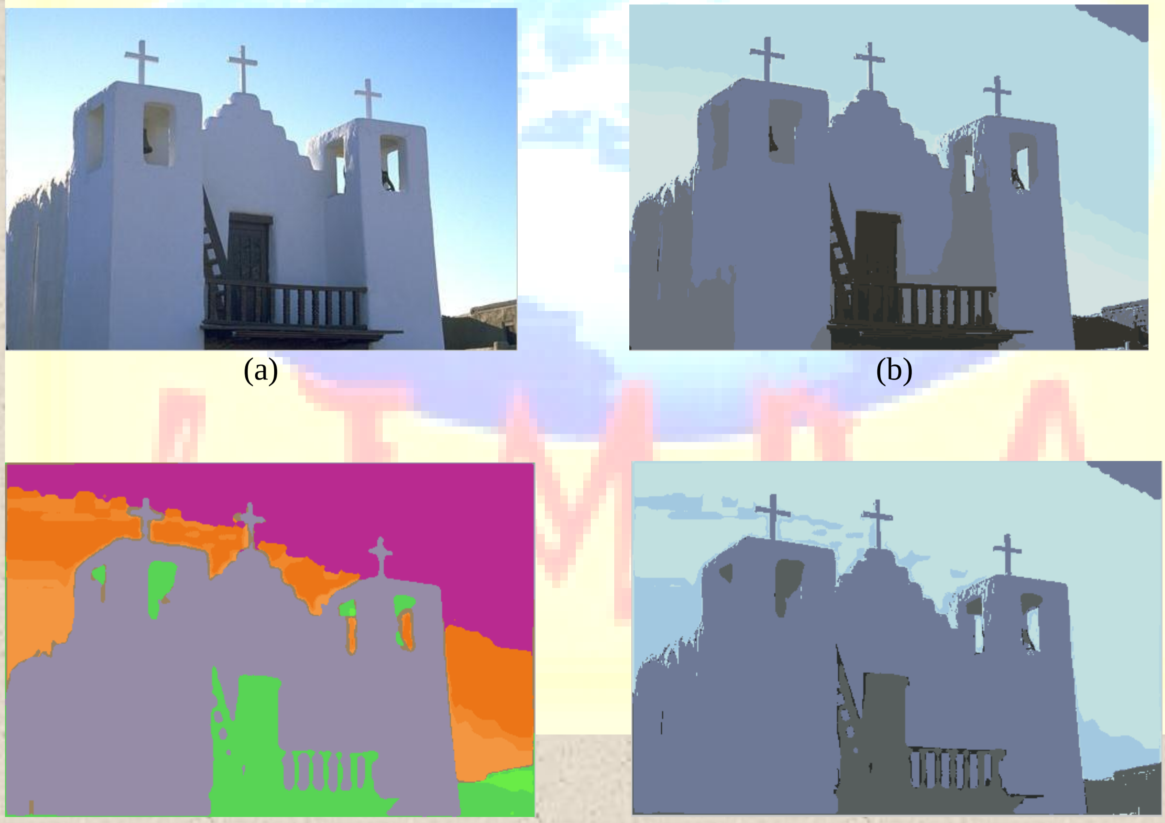
In Eqn. (7), t is the iteration step, $v(t)$ is the neighborhood radius and $L(t)$ is the learning rate. The size of the radius $v(t)$ and the strength of the learning rate $L(t)$ are exponentially reduced with the increase in the number of iterations [see Fig 1(b)–(d)]. The SOM algorithm is iterated until convergence (radius v reaches the size of N_{BMU}) and the final weights of the 2-D network are the dominant colors of the input image.

SELECTION OF THE OPTIMAL NUMBER OF CLUSTERS:

To obtain the optimal number of clusters, a multistep technique that progressively reduces the number of dominant colors resulting after the SOM classification procedure is proposed. In the first step, the pixels in the image are mapped to the final weights of the cells in the SOM network based on the minimum Euclidean distance [see Eqn. (8)]

$$\begin{aligned} &Vj \leftarrow wg, g = \arg\min\|Vj - wi\| \\ &i \in [1,16] \qquad \text{...Eqn. (8)} \\ &j \in [1,n] \end{aligned}$$

The resulting color map can be viewed as a preliminary clustering of the input image.



(a)

(b)

(c)

(d)

Fig 2 : Final Multispace Color Segmentation Results (a) Original image (b) The clustered image in the RGB color space (the number of clusters determined using the SOM procedure is k = 6) (c) The clustered image in the YIQ color space (k = 6) (d) The final multispace color segmentation result. The final number of clusters is 4.

In the second step, a confidence map is constructed where the cumulative smallest distances between the weights of the SOM network and pixels in the image are recorded [see Eqn. (9)]. For all pixels Vj labelled with wi , it is define

$$confidence(wi) = \frac{\sum_{j \in D_{wi}} \|Vj - wi\|}{no\_pixel\_labelled(wi)} \quad \text{.................................Eqn. (9)}$$
$$i \in [1,16]$$

Where Dwi is the image domain is defined by the pixels Vj that are closest to the weights wi .

Table 1: Confidence Map corresponding to Image in Fig 2 (a)

Seed	Confidence Value	No. of samples
C_1	0.072752	17119
C_2	0.099610	14333
C_3	0.077165	18651
C_4	0.103173	2172
C_5	0.070406	17442
C_6	0.081271	16448
C_7	0.075945	17361
C_8	0.201127	2384
C_9	0.066440	16668
C_{10}	0.097796	15595
C_{11}	0.137216	16942
C_{12}	0.105167	19678
C_{13}	0.067716	15011
C_{14}	0.119089	17125
C_{15}	0.0188415	21717
C_{16}	0.130097	9791

The confidence map returns a weighted measure between the variance within the cluster and the number of pixels in the cluster. The lower its value is, the more reliable the estimate *wi* is. The confidence map calculated for the example depicted in Fig. 2.13(a) is shown in Table 2.1. The last step determines the final number of clusters by evaluating the intercluster variability. To achieve this, the similarity matrix is constructed where the Euclidean distances between the weights of any neighboring nodes in the SOM network are stored.

If this distance is smaller than a predefined intercluster threshold, then the node that has the highest confidence value is eliminated. This process is iteratively repeated until the distance between the weights of all adjacent nodes in the SOM network is higher than the predefined threshold value. In this implementation, the intercluster threshold is set to 0.3 and this value proved to be optimal for all images analyzed in this study.

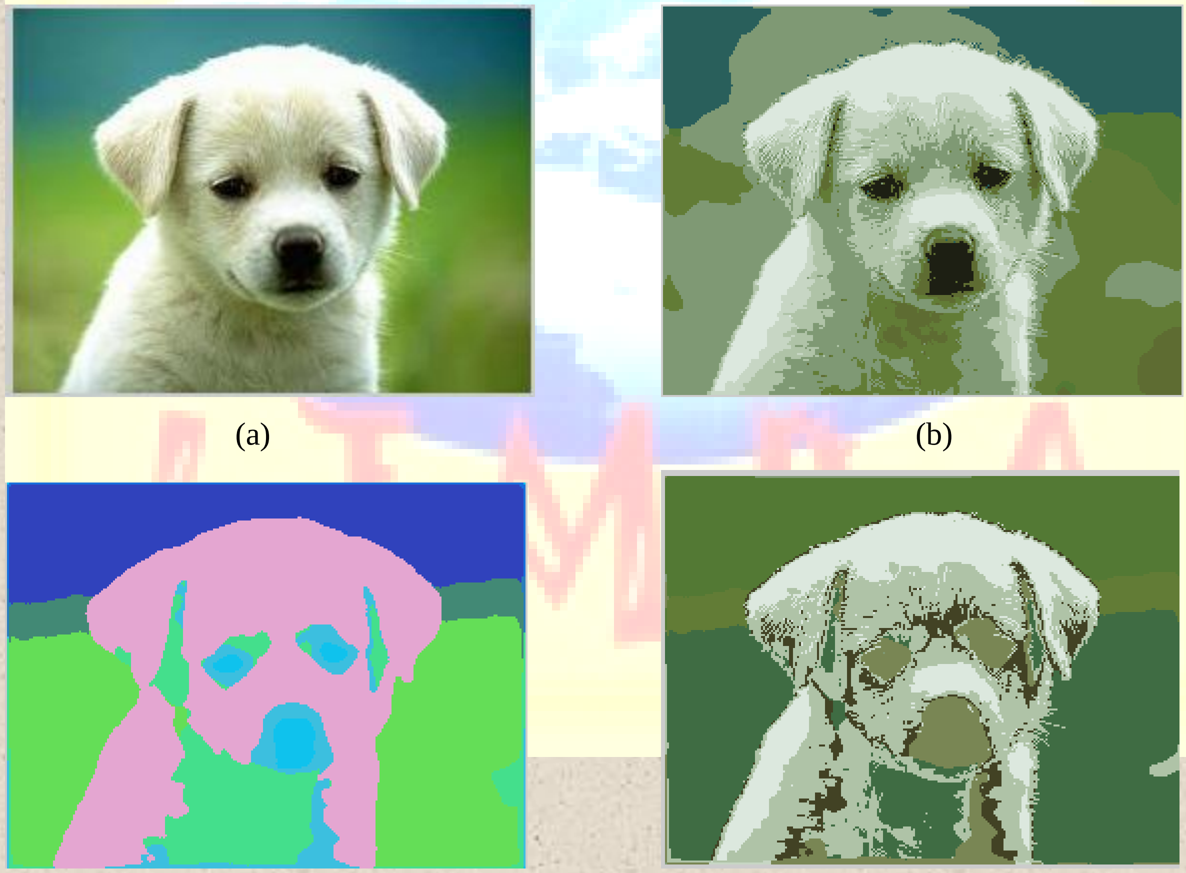
MULTISPACE COLOR SEGMENTATION:

The YIQ image goes through similar operations as the previously analyzed RGB image. Initially, it is filtered using the GB-FAB anisotropic diffusion detailed above and then it is further processed using a K-Means clustering algorithm. The key issue in the extraction of the color features from the YIQ image is the fact that the parameter k that selects the number of clusters for the K-Means algorithm is set to the same value that has been obtained after the application of the SOM procedure to the image represented in the RGB color space. Thus, the parameter k performs the synchronization between the RGB and YIQ channels by forcing the K-Means algorithms applied to the RGB and YIQ images to return the same number of clusters. The dominant colors from the YIQ image that are used to initialize the initial clusters for K-Means algorithm are determined using the same procedure based on color quantization that has been applied to initialize the weights of the SOM network .

The next step of the color extraction algorithm consists in the concatenation of the color features calculated from the RGB and YIQ images where each pixel in the image is defined by a 6-D vector whose components are the R, G, B, Y, I, Q values of the clustered RGB and YIQ images. In the final step, the RGB-YIQ data is clustered with a 6-D K-Means algorithm where the number of clusters is again set to k and the cluster centers are initialized with the dominant colors that were used to initialize the K-Means algorithms that have been applied to cluster the RGB and YIQ images. Fig 2.14 illustrates the performance of the developed multispace color segmentation algorithm when compared to the results obtained when the input image is analyzed in the RGB and YIQ color representations.

It is important to note that, during the multispace partitioning process, some clusters from the initial set may disappear as the clusters become more compact with the increase in the number of iterations (this is achieved by applying a cluster merging procedure that re-labels the adjacent clusters whose centers are close in the RGB-YIQ representation).

At this stage, the approach to analyze the RGB and YIQ images in succession and then fusing the results from the K-means algorithms using multidimensional clustering rather than clustering directly the RGB-YIQ data is adopted. This approach is motivated by the fact that the initialization of the 6-D SOM network using the procedure based on color quantization is unreliable while the 6-D histogram calculated from the RGB-YIQ data is sparse and the peaks are not statistically relevant. This issue circumvents while it attempts to find the optimal result by fusing the clustered RGB and YIQ images which have a reduced dimensionality that is sampled by the parameter k (as opposed to the high dimensionality of the original RGB-YIQ data).


(a) (b)

(c) (d)

Fig 3: Color Segmentation Results (a) Original Natural Images Showing Complex Color-Texture Characteristics (b) RGB Clustered Images (c) YIQ Clustered Images (d) Multispace Color Segmentation Results.

Additional color segmentation results are illustrated in Fig 2.14 when color segmentation algorithm has been applied to natural image with inhomogeneous color-texture characteristics. It can be noted that the shapes of the objects follow the real boundaries present in the original image and the small and narrow objects are not suppressed during the color segmentation process [see the Fig 3]. The segmentation result illustrated in Fig 3 indicate that the color information alone is not sufficient to describe the regions characterized by complex textures in Fig 3. Therefore, the color segmentation with texture features that are extracted using a texture decomposition technique based on Gabor filtering is proposed.

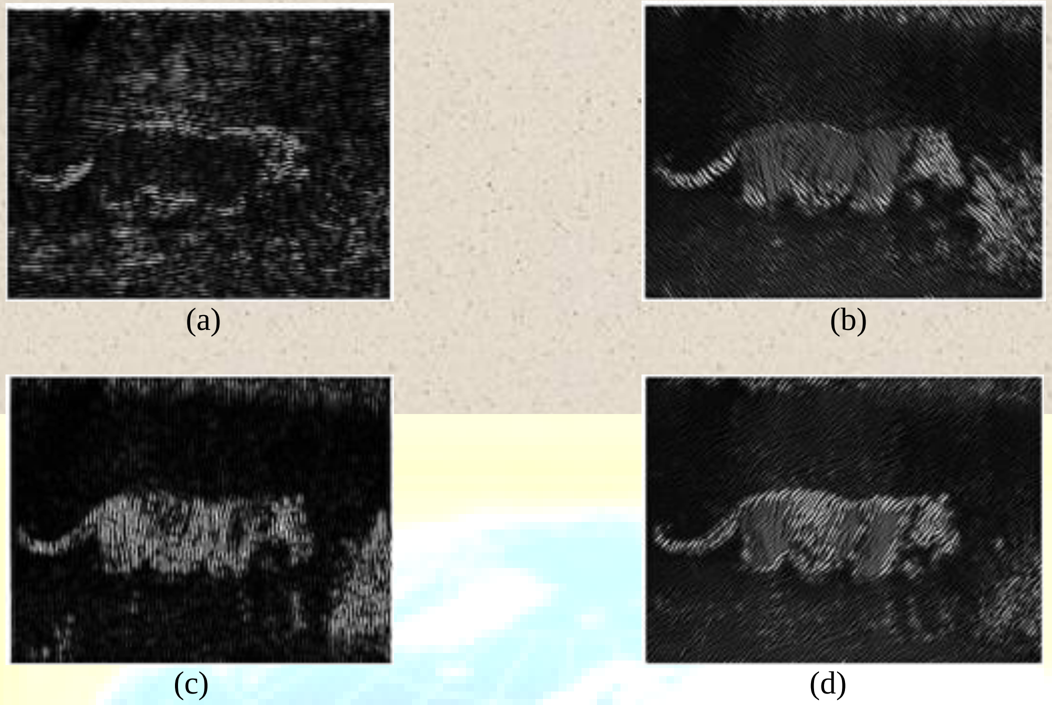
TEXTURE EXTRACTION USING GABOR FILTERS:

There has been a widely accepted consensus among vision researchers that filtering an image with a large number of oriented band pass filters such as Gabor represents an optimal approach to analyze textures. To implement multi channel texture decomposition and is achieved by filtering the input textured image with a 2-D Gabor filter bank that was initially suggested by Daugman and later applied to texture segmentation by Jain and Farrokhnia. The 2-D Gabor function that is used to implement the even-symmetric 2-D discrete filters can be written as

$$G_{\sigma,f,\varphi}(x,y)=\exp\left(-\frac{x'^2+y'^2}{2\sigma^2}\right)\cos(2\Pi fx'+\varphi) \text{ .... Eqn. (10)}$$

Where $x' = x\cos\theta + y\sin\theta$ and $y' = -x\sin\theta + y\cos\theta$

In Eqn.(10), the parameter σ represents the scale of the Gabor filter, θ is the orientation and f is the frequency parameter that controls the number of cycles of the cosine function within the envelope of the 2-D Gaussian (ϕ is the phase offset and it is usually set to zero to implement 2-D even-symmetric filters). The Gabor filters are band pass filters where the parameters σ ,θ and f determine the sub-band that is covered by the Gabor filter in the spatial-frequency domain. The parameters of the Gabor filters are chosen to optimize the trade-off between spectral selectivity and the size of the bank of filters. Typically, the central frequencies are selected to be one octave apart and for each central frequency is constructed a set of filters corresponding to four (0^0 , 45^0 , 90^0 , 135^0) or six orientations (0^0 , 30^0 , 60^0 , 90^0 , 120^0 , 150^0). Fig 4 shows the textures features extracted from a natural image, when the Gabor filters are calculated using four orientations.



(a) (b)

(c) (d)

Fig. 4: Texture Features Extraction using Gabor Filter with Four Orientations: (a) 0^0 (b) 45^0 (c) 90^0 and (d) 135^0

COLOR AND TEXTURE INTEGRATION USING ASKM:

To integrate the color and texture features a spatially adaptive clustering algorithm is proposed. The inclusion of the texture and color features in an adaptive fashion is a difficult task since these attributes are not constant within the image. Thus, the application of standard clustering techniques to complex data such as natural images will lead to over-segmented results since the spatial continuity is not enforced during the space partitioning process. In this project, aim is to develop a space-partitioning algorithm that is able to return meaningful results even when applied to complex natural scenes that exhibit large variations in color and texture. To achieve this new clustering strategy is proposed called ASKM whose implementation can be viewed as a generalization of the K-Means algorithm. The ASKM technique attempts to minimize the errors in the assignment of the data-points into clusters by sampling adaptively the local texture continuity and the local color smoothness in the image. The inputs of the ASKM algorithm are: the color segmented image, the texture images and the final number of clusters k that has been established using the SOM based procedure. The main idea behind ASKM is to minimize an objective function J based on the fitting between the local color and local texture distributions calculated for each data-point (pixel) in the image and the color and texture distributions calculated for each cluster. This approach is motivated by the fact that the color-texture distribution enforces the spatial continuity in the data partitioning process since the color and texture information are evaluated in a local neighborhood for all pixels in the image.

The local color distribution for the data-point at location (x, y) is calculated as follows:

$$H_C^{SxS}(x,y) = \bigcup_{b\in[1,k]} h_C^{sxs}(x,y,b) \text{ ………………………………………………..….Eqn. (11)}$$

Where

$$h_C^{sxs}(x,y,b) = \sum_{p=(x-s/2)}^{x+s/2} \sum_{q=(y-s/2)}^{y+s/2} \delta(C(p,q),b)$$

and

$$\delta(i,j) = \begin{Bmatrix} 1i = j \\ 0i \neq j \end{Bmatrix}$$

Where $H_C^{SxS}(x,y)$ is the local color distribution calculated from the color segmented image C in the neighborhood of size $s \times s$ around the data-point at position(x, y) and k is the number of clusters. In Eqn. (12), the union operator defines the concatenation of the individual histogram bins $h_C^{sxs}(x,y,b)$ b-[1, k], that are calculated from the color segmented image C. The local texture distribution $H_T^{sxs}(x,y)$ is obtained by concatenating the distributions $H_{Tj}^{sxs}(x,y)$ as follows:

$$H_{Tj}^{sxs}(x,y) = \bigcup_{b\in[0,255]} h_{Tj}^{sxs}(x,y,b)$$

$$h_{Tj}^{sxs}(x,y,b) = \sum_{p=(x-s/2)}^{x+s/2} \sum_{q=(y-s/2)}^{y+s/2} \delta[Tj(p,q),b] \text{ ……………………………………………….. Eqn. (12)}$$

$$j \in [1,\alpha]$$

$$H_T^{sxs}(x,y) = \bigcup_{j\in[1,\alpha]} H_{Tj}^{sxs}(x,y)$$

$$= [H_{T1}^{sxs}, H_{T2}^{sxs}, ..., H_{T\alpha}^{sxs} \text{ …………………………………………………………………… Eqn. (13)}$$

Where Tj is the jth Gabor filtered image and α is the total number of texture orientations. The pixel values of the texture images Tj are normalized in the range [0,255].

In order to accommodate the color-texture distributions in the clustering process, the global objective function of the standard K-Means algorithm is replaced with the formulation shown in Eqn. (14). The aim of the ASKM algorithm is the minimization of the objective function J that is composed of two distinct terms that impose the local coherence constraints. The first term optimizes the fitting between the local color distribution for the data point under analysis and the global color distribution of each cluster, while the second term optimizes the fitting between the local texture distributions for the same data point with the global texture distribution of each cluster.

$$J = \sum_{x=1}^{width}\sum_{y=1}^{height}\left\{\sum_{i=1}^{k}\left[\min_{s\in[3x3,\ldots,25x25]} KS(H_C^{sxs}(x,y),H_C^{i}) + \min_{s\in[3x3,\ldots,25x25]} KS(H_T^{sxs}(x,y),H_T^{i}\right]\right\} \text{.............Eqn. (14)}$$

In Eqn.(14), k is the number of clusters, $s \times s$ defines the size of the local window, $H_C^{SxS}(x,y)$ and $H_T^{sxs}(x,y)$ are the local color and the local texture distributions calculated for the pixel at position (x, y) respectively, and are the color and texture distributions for the cluster with index respectively. The similarity between the local color-texture distributions and the global color-texture distributions of the clusters is evaluated using the Kolmogorov–Smirnov (KS) metric

$$KS(Ha,Hb) = \sum_{i\in[0,hist\_size]}\left|\frac{ha(i)}{na} - \frac{hb(i)}{nb}\right| \text{ .... Eqn. (15)}$$

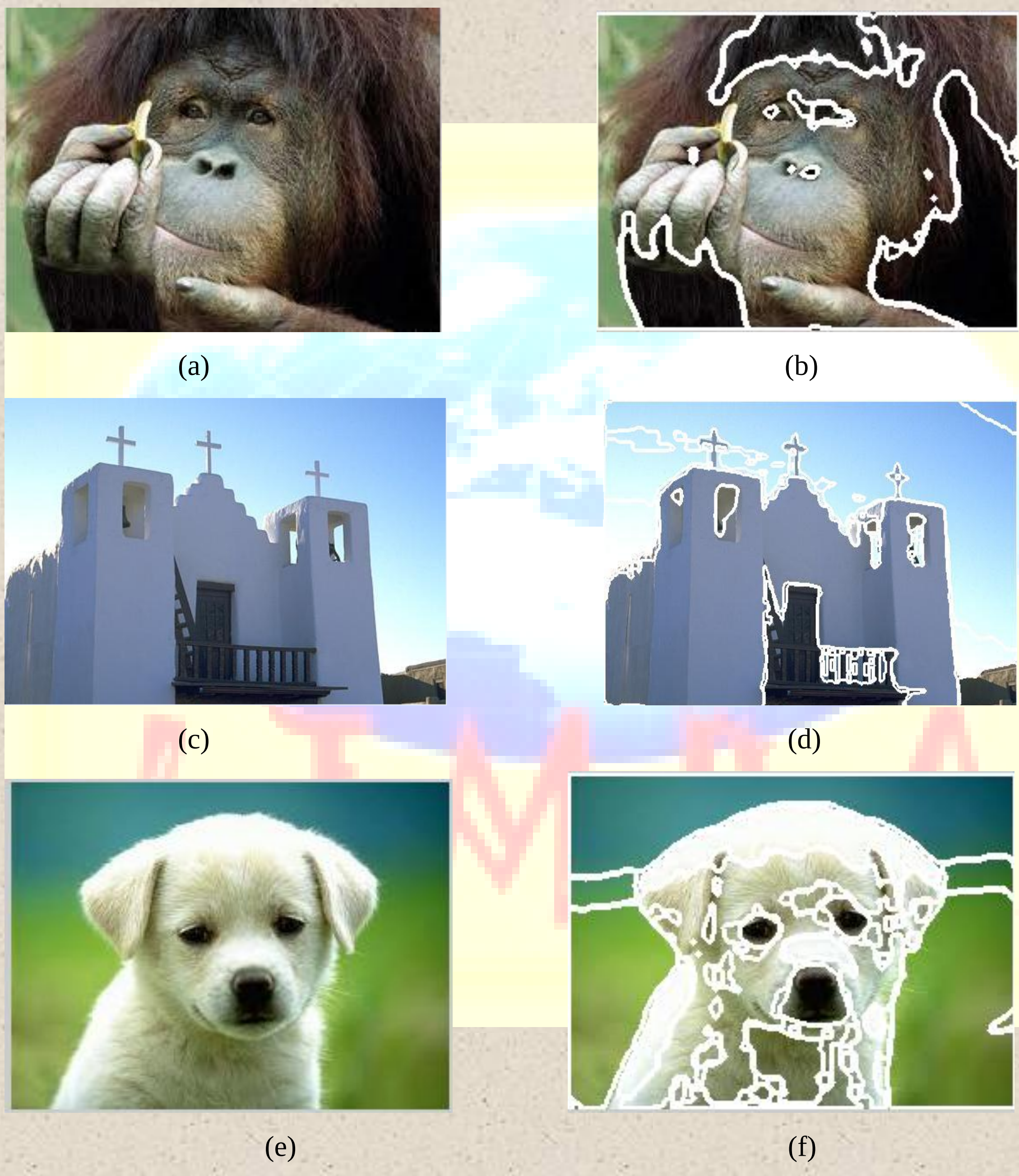
Where na and nb are the number of data points in the distributions Ha and Hb, respectively. The main advantage of the KS metric over other similarity metrics such as G-statistic and the Kullback divergence is the fact that the KS metric is normalized and the result of the comparison between the distributions and is bounded in the interval [0, 2]. The fitting between the local color-texture distributions and global color-texture distributions of the clusters is performed adaptively for multiple window sizes in the interval 3×3 to 25×25. The evaluation of the fitting between the local and global distributions using a multi resolution approach is motivated by the fact that the color composition of the texture in the image is not constant and the algorithm adjusts the window size until it is achieved the best fit value. It is important to note that the global color-texture distributions and are updated after each iteration and the algorithm is executed until convergence.

RESULTS AND CONCLUSION:

This test was conducted on natural image datasets and the results were qualitatively and quantitatively evaluated.

EXPERIMENTS PERFORMED ON NATURAL IMAGES :

The performance of proposed system for color texture segmentation has been finally tested over a set of real images, which have been generally extracted from the test images. Natural scenes with animals predominates these images, since nature is the most complex and rich source of colors and textures.


(a) (b)

(c) (d)

(e) (f)

Fig 5 : Segmentation of Natural Images. (a), (c) and (e) Original Images
(b), (d) and (f) Segmentation Results.

Some segmentation results obtained using this system. Meaningful regions in images are successfully detected and the usefulness of CTex is demonstrated. Fig 5 (b) shows the segmentation of monkey. The monkey is correctly segmented and moreover, although the animal is absolutely black several parts of its skin are identified due to their different textural properties. Similar situations occur with other images in which animals are present. In the image with dog region at eyes and mouth are segmented as different regions. It is true that in these cases many of human would group all these regions to compose a single region related to the whole animal body. Nevertheless, this process of assembling is more related to the knowledge that about animals that to the basic process of segmentation. Hence the segmentation performed by this CTex system is correct as it distinguishes regions with different color texture. The correctness of boundaries obtained in these segmentations is also shown by the sketch of detected borders over original images.

EXPERIMENTS PERFORMED ON FRUIT IMAGES:

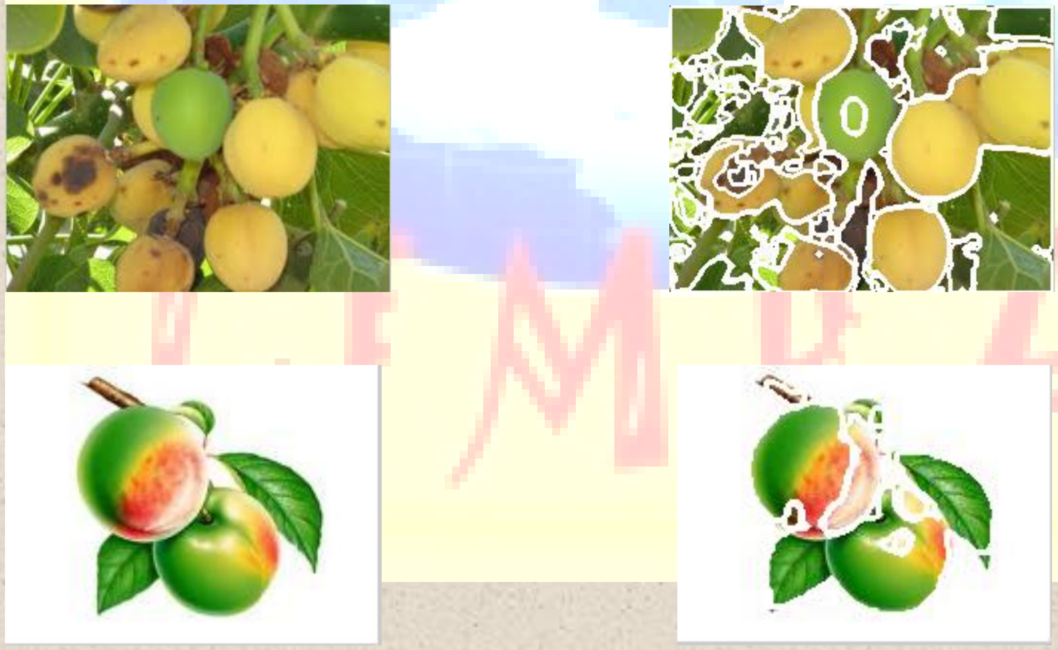


Fig 6: Segmentation of Fruit Images

The proposed CTex segmentation algorithm is tested on a large number of complex natural images in order to evaluate its performance with respect to the identification of perceptual color-texture homogenous regions. To achieve this goal, CTex technique is applied to natural images databases that include images characterized by non uniform textures, fuzzy borders, and low image contrast. The experiments were conducted to obtain qualitative evaluation of the performance of the CTex color-texture segmentation framework. This is shown in Fig 6.

CONCLUSION:

In this paper, a new segmentation algorithm is presented where the color and texture features are adaptively evaluated by a clustering strategy that enforces the spatial constraints during the assignment of the data into image regions with uniform texture and color characteristics. The main contribution of this work resides in the development of a novel multispace color segmentation scheme where an unsupervised SOM classifier was applied to extract the dominant colors and estimate the optimal number of clusters in the image. The second strand of the algorithm dealt with the extraction of the texture features using a multichannel decomposition scheme based on Gabor filtering. The inclusion of the color and texture features in a composite descriptor proved to be effective in the identification of the image regions with homogenous characteristics. The performance of the developed color-texture segmentation algorithm has been qualitatively evaluated on a large number of natural images and the experimental results indicate that CTex algorithm is able to produce accurate segmentation results even when applied to images characterized by low resolution and low contrast.

REFERENCES:

- Dana E. Ilea and Paul F. Whelan, IEEE,"CTex—An Adaptive Unsupervised Segmentation Algorithm Based on Color-Texture Coherence", IEEE Transactions on Image processing, Vol.17, No.10, October 2008.
- Xavier Munoz Pujol, "Image Segmentation Integrating color, texture and boundary information", PhD Thesis, Girona University, December 2002.
- Y. Deng and B. S. Manjunath, "Unsupervised segmentation of color texture regions in images and video," IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 8, pp. 800–810, Aug. 2001.
- J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz, "Adaptive perceptual color-texture image segmentation," IEEE Trans. Image Process., vol. 14, no. 10, pp. 1524–1536, Oct. 2005.

- P. Jeyanthi1 and V. Jawahar Senthil Kumar "Image Classification by K-means Clustering" ISSN 0973-6107 Volume 3 Number 1, pp. 1–8, 2010.
- M. Mirmehdi and M. Petrou, "Segmentation of color textures," IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 2, pp. 142–159, Feb. 2000.
- T. Kohonen, "Self-Organizing Maps", Springer Ser. Information Sciences, 3rd ed. New York: Springer Verlang, vol.30, 2001.
- D. E. Ilea and P. F. Whelan, "Automatic segmentation of skin cancer images using adaptive color clustering," China-Ireland Int.Conf. Information and Communications Technologies, Hangzhou, China, pp. 348–351, 2006. http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html
- Vision Texture (VisTex) Database, Media Lab., Mass. Inst. Technol., Cambridge. http://www.outex.oulu.fi.Outex Natural Images Database.