

An OpenCV Implementation of Supervised Texture Segmentation Using Gabor Filters.

Rushil Anirudh (ranirudh@asu.edu), Suhas Ranganath (srangan8@asu.edu).

Abstract—In this paper we first study the basic principles of texture, its classification and properties. We begin this paper by studying the current literature in this field and take a brief look at the proposed theories. We follow this up by showing the results for the implementation of an important paper in the field of texture segmentation and classification. The paper - "Unsupervised Texture Segmentation Using Gabor Filters" The results shown by Anil K Jain and Farshid Farrokhnia in their path breaking research paper went on to open new areas of research in texture and made a huge leap forward in our understanding of its properties.

I. INTRODUCTION

TEXTURE describes a visual information which is related to local spatial variations of color, orientation and intensity in an image. It is usually described by qualitative adjectives such as smooth or rough, coarse or fine, homogeneous or random, etc. This information is fundamental in image analysis and interpretation and the segmentation of an image into homogeneous regions, in terms of textural features, remains a complex issue. An effective and efficient texture segmentation method is of key interest in numerous domains such as biomedical image analysis, industrial inspection, analysis of remote sensing images, sonar or aerial imagery etc[1]. Petrou and Sevilla [7] give two reasons for studying texture namely:

- Texture may be a nuisance in an automatic vision system. For example, if we were to recognise an object from its shape, texture would create extra lines in the edge map of the object and the shape recognition algorithm would be confused.
- Texture may be an important cue in object recognition as it tells us something about the material from which the object is made. For example, it can be used to discriminate a city from the woods and the fields etc.

The problems associated with texture are that it depends on scale, illumination, rotation etc. Similar textures observed under different lighting conditions and/or at different scales can appear to be vastly different hence confusing the segmentation algorithm. It is therefore of prime importance to develop robust algorithms which can take into consideration all of the mentioned problems to classify and segment texture accurately.

The broad outline for any texture segmentation process (as for any pattern recognition system) would be to first identify the important characteristic features and cluster them. Existing literatures approach the problem of texture segmentation mainly in four different ways:

- Use of Statistical Properties like Co-occurrence Matrices.
- Use of Geometric Properties like Fractals.
- Feature Extraction using Signal Processing, Gabor Filters & Wavelets.
- Use of Suitable Texture Models like Gaussian Markov Fields.
- Fusion of two or more features from the above methods.

In the following sections we will review the important papers in the aforementioned fields and discuss their theories briefly.

II. USE OF STATISTICAL PROPERTIES

One of the first approaches to classify textures was using statistical properties as described by Haralick et al.[2] In the seminal paper, Haralick et al discuss the close relationship between tones and textures. It is observed that, when a small-area patch of an image has little variation i.e., little variation of features of discrete gray tone the dominant property of that area is tone. When a small-area patch has a wide variation of features of discrete gray tone, the dominant property of that area is texture. Their method to identify textual features assumes the fact that most of the textual information is contained in a set of gray tone spatial dependence matrices which are computed for various angular relationships and distances between neighboring resolution cellpairs on the image.

The first step is to compute co-occurring probabilities of all pairwise combinations of quantized (G) grey levels (i,j) in the fixed size (N) spatial window given two parameters: inter-pixel distance (δ) and orientation (θ), i.e.

$$P(i, j) = Pr(i, j | \delta, \theta, G, N). \quad (1)$$

Usually a variety of orientations and inter-pixel distances are selected. While a coarser quantization accelerates calculations, it also leads to loss of textual features. The next step is to apply statistics to the co-occurring probabilities. Statistics that identify some structural aspect of the arrangement are used to generate texture features. Some of the statistics used are angular second-momentum, entropy, sum-entropy, difference-entropy etc.

The disadvantage of this theory is that an optimal size of the subregion (N) and the inter-pixel distances are not mentioned. Small windows can lead to poor local estimates and large windows increase the risk of multiple textures appearing in the window which produces misleading features.

III. USE OF SPATIAL AND GEOMETRIC PROPERTIES

A. Using Fractals:

It has been said that most natural surfaces are spatially isotropic fractals[3] and since the intensity images of these

surfaces are also fractals, fractal analysis has been successfully applied in several fields of image processing. It offers the potential of unifying and simplifying various two dimensional texture descriptions, as well as the possibility of interpreting them in terms of three dimensional structure of the image. Multifractal analysis using box-counting based multifractal dimension estimation are quite popular amongst the image segmentation community. However even inspite of its computational efficiency most of the time, results are less accurate than desired.

Xia et al [4] propose a novel multifractal estimation algorithm based on mathematical morphology. A new set of multifractal descriptors namely, local morphological multifractal exponents, are defined to perform texture analysis. Here they have used a series of structural elements of different scales are used to measure the image surface. These structural elements along with an iterative dilation scheme are used so that computational complexity of morphological operations is tremendously reduced. This novel algorithm delicately avoids the drawback of various box-counting methods and thus achieves better accuracy in characterizing the local scaling properties of a texture image.

B. Using Local Binary Patterns (LBPs):

LBPs were first identified by Ojala et al [5] in which they use them to identify and distinguish between textures. A 3x3 window is considered over the image and its elements are thresholded by the value of the center pixel. The values of the pixels in the thresholded neighborhood are multiplied by the weights given to the corresponding pixels. Finally, the values of the eight pixels are summed to obtain the number of this texture unit. The LBP method is a gray-scale invariant and can be easily combined with a simple contrast measure by computing for each neighborhood the difference of the average gray level of those pixels which have the value 1, and those which have the value 0, respectively. These results were further improved by Ojala et al.[6] to make it more robust in terms of gray scale and rotation invariance. The modified LBP now considered a general area of radius R (specified by the user) instead of the classical 3x3 window search method. It was also shown that the response of this operator over the region of an image was a very powerful texture feature.

The main advantage of using this operator is that the combination of statistical and structural properties, we can detect a large number of microstructures (edges, lines, spots and flat areas). It is computationally simple and has rotation and scale invariance.

IV. FEATURE EXTRACTION BY WAVELETS, GABOR FILTERS :

A. Using Wavelets:

Charalampidis and Kasparis [8] have developed a wavelet based approach to classify and segment textures. They introduce a feature set which is based on an extension of Fractal Dimension (FD) features. Most of the literature in traditional FD analysis assumes that natural textures exhibit similar roughness

over large number of scales. This assumption fails for many textures. Charalampidis and Kasparis' feature set extracts roughness information taking into account all the single-scale features and multiple-scale features for a complete textural representation. Wavelets are used due to their ability to extract information at different resolutions. Features are extracted in multiple directions and the feature vector is made rotational invariant retaining directional information. An iterative -means scheme is used for segmentation, and a Bayesian classifier is used for classification. The use of the roughness feature set results in high-quality segmentation performance and classification performance. The roughness feature vector results in good segmentation, as well as better classification performance. In addition of retaining retain the important characteristics of fractal-based methods; they are also insensitive to rotation. The percentage of correct classification is relatively high when the images are skewed and considerably better than the Hurst features.[9]

B. Using Gabor Filters:

A Gabor function is a Gaussian modulated complex sinusoid in the spatial domain. Gabor Filters are used in multi-channel filtering techniques for texture segmentation. This filter set forms an approximate basis for the wavelet transform with the gabor function defined as the wavelet. Frequency and orientation representations of Gabor filters are similar to those of the human visual system, and they have been found to be particularly appropriate for texture representation and discrimination[10]. Since we will be implementing a Gabor Filter based approach for texture segmentation, we will study it in detail here.

The response of an even symmetric Gabor filter is defined as:

$$h(x, y) = \exp\left(-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right)\cos(2\pi u_0). \quad (2)$$

where u_0 is the frequency of a sinusoidal plane wave along the x -axis (i.e the 0° Orientation), and σ_x and σ_y are the space constants of the gaussian envelope along the x and y axes, respectively.

One of the most popular texture segmenting methods was suggested by Jain et al [11] which obtained the response of the texture after filtering it through different orientations and then extracted Textual Features for segmentation. They used a modification of Hubert Γ statistics as a relative index to estimate the "true" number of texture categories making it an unsupervised texture segmentation algorithm.

Newer methods in Literature which use Gabor Filters perform segmentation with the combination of Gabor Filters and a feature set derived from one of the earlier mentioned theories. One popular method was proposed by Clausi and Deng [12] which took into consideration Gabor Filters and Co-occurrence probabilities. This fused feature set The fused feature set utilizes both the Gabor filters capability of accurately capturing lower and mid-frequency texture information and the GLCPs capability in texture information relevant to higher frequency

components. The main work of this paper lies in the fact that features are not fused blindly, but a rationale is provided here for the fusion of these particular features. The fusion is based on the theoretical analysis and experimental verification of each method so as to combine robust, reliable, and complementary features. A combined feature set provides a much more powerful texture segmentation approach, as expected.

V. USING SUITABLE TEXTURE MODELS LIKE MARKOV RANDOM FIELDS:

Pedro & Sevilla [7] define a Markov Random Field (MRF) as a random field which possesses the Markovian Property: the value of a pixel depends directly only on the values of the neighbouring pixels and on no other pixel. To characterise textures, we model them as MRFs. We have to choose first the type of neighbourhood we shall adopt and then the form of the probability density function of the random process. Once these are fixed the parameters on which the pdf depends on, characterise the texture. These are called Markov Parameters. Zheng et al [13] talk about an adaptive segmentation algorithm based on these MRFs. The model they propose has two dependent components: one models the observed image to estimate features, and the other models the labels for segmentation. The homogeneity of the sub image is obtained by using only the pixels having same labels as the same pattern. With the acquired features, the labeling is obtained solving a maximum a posteriori problem. The feature set and the labeling are mutually dependent on each other and are alternately optimized. As the accuracy improves, the labeling detects the boundary of each texture pattern adaptively. While traditional feature-based texture segmentation methods usually suffer from the inaccuracy, mainly caused by assuming that each textured sub image used to estimate a feature is homogeneous. The proposed approach can differentiate textured images more accurately through adaptive processes.

The features of each pattern are assumed to have a Gaussian distribution due to mathematical tractability. This assumption is, however, a shortcoming in the proposed approach as those images whose features defy the Gaussian Distribution, will fail to be segmented correctly.

VI. IMPLEMENTATION OF TEXTURE SEGMENTATION USING GABOR FILTERS

In the following sections we will study the important theories put forward by Anil K. Jain and Farshid Farrokhnia [11]. Subsequently we will show our implementation results and attempt to discuss improvements.

In this paper, the authors focus on a multi-channel filtering approach using Gabor Filters. This is intuitively appealing because it allows us to exploit the differences in dominant sizes and orientations of different textures. While most other approaches for analysis have to be extended to accommodate this paradigm, this approach is inherently multi-resolutional.

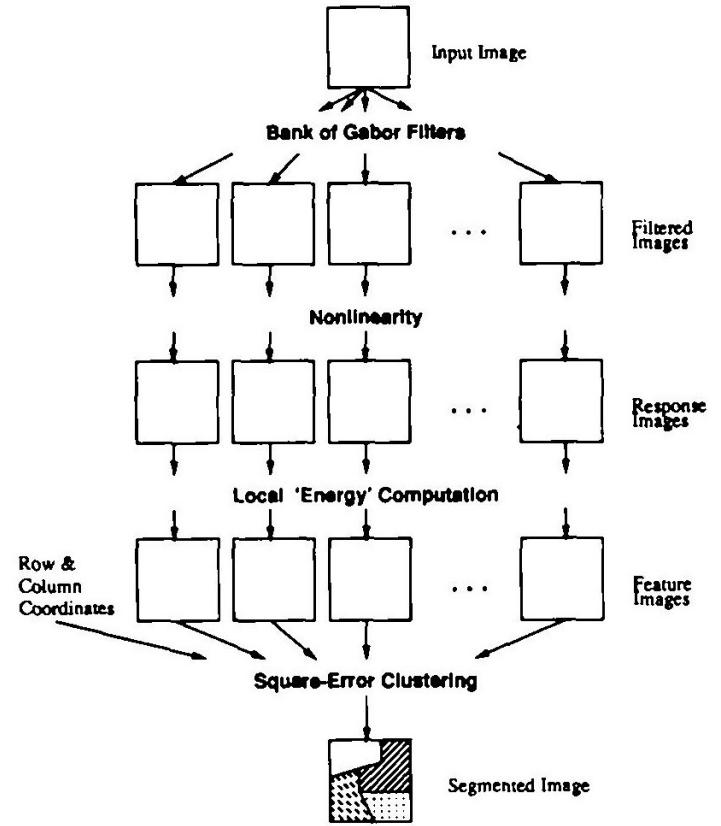


Fig. 1. An Overview of the Texture Segmentation Algorithm. Source: [11]

VII. BRIEF LAYOUT:

A multichannel filtering technique is presented that uses a bank of even-symmetric Gabor Filters. A systematic filter selection scheme is proposed which is based on the reconstruction of the input image from the filtered images. Each (selected) filtered image is subjected to non linear transformation that behaves as a blob detector. The combination of multi-channel filtering and the non linear stages can be viewed as performing multi-scale blob detection.

The frequency response of Gabor Filters in the space domain has been defined earlier, however the frequency and orientation selective properties become more explicit in the Frequency Domain as shown below:

$$H(u, v) = A \left(\exp \left\{ -\frac{1}{2} \left[\frac{(u - u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2} \right] \right\} + \exp \left\{ -\frac{1}{2} \left[\frac{(u + u_0)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2} \right] \right\} \right) \quad (3)$$

where $\sigma_u = 1/2\pi\sigma_x$ and $\sigma_v = 1/2\pi\sigma_y$ and $A = 2\pi\sigma_x\sigma_y$. The Fourier Domain representation in (3) specifies the amount by which the filter modifies or modulates each frequency component of the input image. Such representations are, therefore, referred to as modulation transfer functions (MTF).

A. Choice of Filter Parameters:

For orientations θ_0 are used: $0^\circ, 45^\circ, 90^\circ, 135^\circ$. For an image array with a width of N_c pixels, where N_c is a power

of 2, the following values of radial frequency u_0 are used:

$$1\sqrt{2}, 2\sqrt{2}, \dots, N_c/4\sqrt{2}$$

Note that the radial frequencies are one octave apart. Several experiments have shown that the frequency bandwidth of simple cells in the visual cortex is about 1 octave.

The total number of Gabor Filters in the filter set is therefore given by $4\log_2(N_c/2)$. For example, a 256x256 image a set of 28 filters can be used : 4 orientations and 7 radial frequencies.

The filtering operations using the filter set can be interpreted as computing the wavelet transform of the input image at selected spatial frequencies.

B. Filter Selection

A systematic filter selection scheme based on the least square errors is used so that only a subset of the filtered images can be used to reduce computational complexity. Let $s(x,y)$ be the reconstruction of the input image obtained by adding all the filtered images. We assume that this is a good approximation of the original image (See Results). Let $\hat{s}(x,y)$ be the partial reconstruction of the image obtained by adding a given subset of the filtered images. The error involved in using $\hat{s}(x,y)$ is given by:

$$SSE = \sum_{x,y} [\hat{s}(x,y) - s(x,y)]^2$$

A fraction of the intensity variations in $s(x,y)$ that is explained by $\hat{s}(x,y)$ can be measured by the coefficient of determination (COD):

$$R^2 = 1 - \frac{SSE}{SSTOT}$$

where $SSTOT = \sum_{x,y} [s(x,y)]^2$

The best set of filtered images are determined by sorting their energies in decreasing order and then picking the first 'n' images which have a combined energy of ≤ 0.95 times the original image energy. In other words, until $R^2 \leq 0.95$. Calculating the least square error tends to be computationally expensive, therefore we used the energy method to select the "best set" of filtered images.

VIII. COMPUTING THE FEATURE IMAGES:

An important goal in texture analysis is to develop textual features that can help in discriminating textures. The following procedure is used to compute feature images from each filtered image. First, each filtered image is subjected to a non linear transformation:

$$\psi(t) = \tanh(\alpha t) = \frac{1 - e^{-2\alpha t}}{1 + e^{-2\alpha t}} \quad (4)$$

Where $\alpha = 0.25$ was used in accordance to the paper. This results in rapidly saturating, threshold-like transformation. The next step formally calculates the feature image $e_k(x,y)$ given by:

$$e_k(x,y) = \frac{1}{M^2} \sum_{a,b \in W_{x,y}} |\psi(r_k(a,b))|, \quad (5)$$

where $\psi(.)$ is the non linear transformation as in (4) and $W_{x,y}$ is the MxM window centered at the pixel (x,y). The size of the window is an important parameter as more accurate localization can be achieved by using smaller window sizes while better texture characterization can be achieved by using larger window sizes. We have used a window size of M=8 in our experiments. We have also used Gaussian weighted windows, which result in more accurate localization of texture boundaries. For each window, we use a Gaussian Window whose space constant σ is given by

$$\sigma = 0.5 \frac{N_c}{u_0} \quad (6)$$

Where N_c and u_0 are defined as before.

IX. INTEGRATING FEATURE IMAGES:

Having obtained the Feature Images, we now need to identify the texture categories. Assuming we have K textures, we can use the feature images as an N-Dimensional Data and cluster it according to the number of textures, i.e. K clusters. One point to note is that a segmentation algorithm that clusters based only on pixel intensity values suffers from an important shortcoming – it does not utilize the spatial (contextual) information. In texture segmentation, pixels next to each other are most likely to belong to the same texture category. This enhances the segmentation process.

A. Deviation from the Clustering Approach in the Paper

The method suggested for clustering is the CLUSTER program[14] which consists of two phases. Phase 1 creates a sequence of clusterings containing $2, 3, \dots, k_{max}$ clusters (i.e K-Means), where k_{max} is specified by the user. Phase 2 then creates another set of clusterings by merging existing clusters two at a time to see if better clusterings can be obtained. After each pass through Phase 1 and Phase 2, the square errors of the clusterings are compared to the square errors of the clusterings that existed before the pass. If any of the square errors are smaller than before, another pass through phases 1 and 2 is initiated until square error cannot be decreased.

In this implementation we have chosen to cluster by using only phase 1 i.e K-Means approach, since this is for the purpose of demonstration of the texture segmentation algorithm. However, better clustering can be achieved by using both Phases 1 & 2.

The authors continue to address the problem of Unsupervised texture segmentation by identifying the optimal number of textures in an image, but we have assumed that it will be User-Specified to the program.

X. RESULTS

The results at different stages are shown here. After clustering the different texture categories, we show the a texture specified by the user in these sample images.



Fig. 2. Baboon and two of textured images.



Fig. 3. Another example with a Natural Scene.



Fig. 4. Examples of Filtered Images at $u_0 = \sqrt{2}$ cycles image/width and $\theta_0 = 90^\circ$

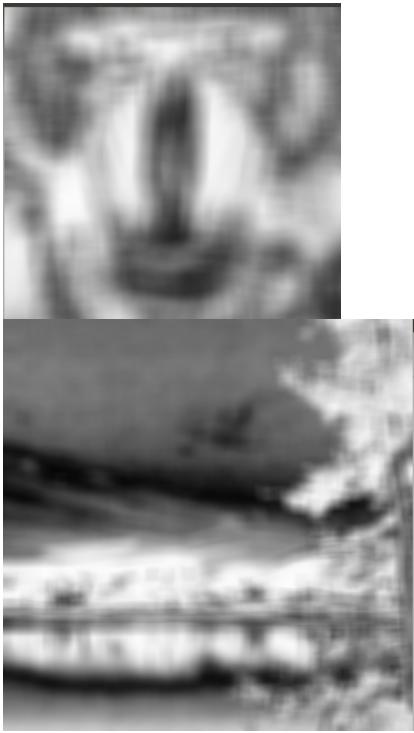


Fig. 5. Feature Images after the Non Linear Transformation
Here shown for $u_0 = \sqrt{2}$ cycles image/width and $\theta_0 = 90^\circ$



Fig. 6. Reconstructed Images (in grayscale) from the Filters are a good approximation of the Original.

XI. FUTURE WORK

We have discussed a classical approach in the areas of image segmentation. There have been several advancements in this field which have made texture segmentaion more common amongst several computer vision systems. There are several areas of work that still need to be worked upon like that of textures in motion like water, smoke etc in videos. Another area of focus is on integration of colour and texture features for enhanced results[15]. Researchers are also investigating segmentation based on visual perception using genetic programming methods [16]. Texture segmentation based on Fuzzy algorithms are an another active area of research [17] [18].

XII. CONLCUSION

In this paper, we defined and studied the basic aspects of texture segmentation and some saw recent literature in this field. Texture Segmentation can be approached in several ways as discussed, like by using feature extraction techniques with the help of Gabor Filters, Wavelets; by statistical measures like Co-occurrence matrices; By using the spatial and geometrical properties like Fractal Dimensions and Linear Binary Patterns or a combination of two or more of these. We then implemented a supervised version of the texture segmentation algorithm proposed by A.K. Jain and F. Farrokhnia. An impediment faced by us during this implementation was to visualize the segmented textures as distinct areas in the image. Although the algorithm successfully segments the textures, a better way to view the different textures can make this implementation more suitable for general demonstration.

REFERENCES

- [1] Karoui, I.; Fablet, R.; Boucher, J.; Augustin, J., *Variational Region-Based Segmentation Using Multiple Texture Statistics*, *IEEE Trans. Image Process.*, Vol.19, NO.12, pp.3146-3156, Dec 2010.
- [2] Haralick, Robert M.; Shanmugam, K.; Dinstein, Its'Hak; *"Textural Features for Image Classification,"*Systems, Man and Cybernetics, IEEE Transactions on , vol.3, no.6, pp.610-621, Nov. 1973.
- [3] A. P. Pentland, *Fractal based description of natural scenes*, IEEE Trans. Pattern. Anal. Mach. Intell., vol. PAMI-6, no. 6, pp. 661674, Jun. 1984.
- [4] Yong Xia; Dagan Feng; Rongchun Zhao; , *"Morphology-based multifractal estimation for texture segmentation,"* Image Processing, IEEE Transactions on , vol.15, no.3, pp.614-623, March 2006
- [5] Timo Ojala, Matti Pietikäinen, David Harwood, *A comparative study of texture measures with classification based on featured distributions*, *Pattern Recognition*, Volume 29, Issue 1, January 1996, Pages 51-59, ISSN 0031-320.
- [6] Ojala, Timo and Pietikäinen, Matti and Mäenpää, Topi; *Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns*;IEEE Trans. Pattern Anal. Mach. Intell., Vol 24 Issue 7, pp 971-987. July 2002.
- [7] Maria Petrou, Pedro Garcia Sevilla; *Dealing with Texture*; John Wiley & Sons, Ltd. 1st ed,2006.
- [8] Charalampidis, D.; Kasparis, T.; , *"Wavelet-based rotational invariant roughness features for texture classification and segmentation,"* Image Processing, IEEE Transactions on , vol.11, no.8, pp. 825- 837, Aug 2002.
- [9] Kaplan, L.M.; Murenzi, R.; , *"Texture segmentation using multiscale Hurst features,"* Image Processing, 1997. Proceedings., International Conference on , vol.3, no. , pp.205-208 vol.3, 26-29 Oct 1997.
- [10] en.wikipedia.org/wiki/gabor_filter, Wikimedia Foundation, Inc. 2011.
- [11] Anil K. Jain, Farshid Farrokhnia, *Unsupervised texture segmentation using Gabor filters*, *Pattern Recognition*, Volume 24, Issue 12, 1991, Pages 1167-1186.
- [12] Clausi, D.A.; Huang Deng; , *"Design-based texture feature fusion using Gabor filters and co-occurrence probabilities,"* Image Processing, IEEE Transactions on , vol.14, no.7, pp.925-936, July 2005.

- [13] Lei Zheng; Liu, J.C.; Chan, A.K.; Smith, W.; , "Object-based image segmentation using DWT/RDWT multiresolution Markov random field," Acoustics, Speech, and Signal Processing, 1999. ICASSP '99. Proceedings., 1999 IEEE International Conference on , vol.6, no., pp.3485-3488 vol.6, 15-19 Mar 1999.
- [14] A.K. Jain and R.C Dubes, *Algorithms for Clustering Data*; Prentice-Hall, Englewood Cliffs, NJ (1998).
- [15] Ilea, Dana E. Whelan, Paul F. *Image segmentation based on the integration of colour-texture descriptors-A Review* ; Pattern Recognition In Press. Corrected Proof, 2010.
- [16] Song, Andy and Ciesielski, Vic *Texture Segmentation by Genetic Programming* NYT Press, Evolutionary Computation; Dec 2008.
- [17] Xian Yi,Liu Haihua,Liao Qiuping; *Weighted Fuzzy C-Means Algorithm Based on Multi-feature for Texture Image Segmentation*, CNKI - Modern Scientific Instruments, January 2011.
- [18] G. Uma Maheswari, K. Ramara, D. Manimegalaia and V. Gomathia ; *An adaptive region based color texture segmentation using fuzzified distance metric.*; Applied Soft Computing Volume 11, Issue 2, March 2011, Pages 2916-2924