

Texture Analysis of a Color Image Using Traditional and Circular Gabor Filters

N. Sasikala, G. Shruthi

¹Associate Professor, Department of Electronics & Communication Engineering, Kamala Institute of Technology & Science, Singapur, Huzurabad, Karimnagar, India

²Assistant Professor, Department of Electronics & Communication Engineering, Kamala Institute of Technology & Science, Singapur, Huzurabad, Karimnagar, India

Abstract— According to computer vision, segmentation is defined as the process of partitioning a digital image into multiple segments, where multiple segments are sets of pixels, in other words super pixels. Main objective of segmentation is to change and, or simplify the representation of a digital image into something that is much more significant and easier to analyze. Objects and boundaries like lines, curves, etc. in images can be normally located by using image segmentation. More accurately, the process of assigning a tag to every pixel in an image such that pixels with the same label share specific visual characteristics is known as image segmentation. The outcome of image segmentation is a set of surface (especially of a curving form) extracted from the image, a set of segments that as a group cover the entire image. In a segment every pixels are similar with regard to computed property or some characteristic, such as intensity, texture, or color. A Gabor filter is a linear filter used for edge detection in image processing which is named after Dennis Gabor. Gabor filter frequency and orientation representations are similar to those of human visual system, for texture representation and discrimination it has been found to be remarkably appropriate. Gabor filter is a powerful tool in texture analysis. Traditional Gabor function (TGF) represents a Gaussian function modulated with the help of an oriented complex sinusoidal signal.

Keywords— Color image, Gabor filters, Traditional Gabor function.

I. INTRODUCTION

Neighbouring segments are considerably different with regard to the same characteristics. Image segmentation can also be considered as partition of an image into set of non overlapping areas whose combination is the complete image, few rules to be followed for regions resultant from the image segmentation can be confirmed as:

- All segments should be uniform and homogeneous with regard to some characteristics.
- Region's interiors should be uncomplicated and without many small holes.
- Neighbouring segments should have significantly different values with regard to the characteristic according which they are uniform.
- Every segment's boundaries should be simple, not blurring, and must be spatially accurate.

A grey – scale or grey – scale digital image is an image in computing and photography in which a single sample is the value of each pixel, i.e., it holds only intensity information. Images of this type are collected exclusively of grey shades, varies to white at the strongest from black at the weakest. Images are with two colours only, black and white also known as bi – level or binary images in the context of computer imaging and grey scale images are different for these one – bit images. Grey – scale images contain many grey shades in between. Due to the absence of any chromatic variation (no colour), grey – scale images are also known as monochromatic.



Fig. 1.1 : Gray – Scale Lightening

Let the Fig 1.1 be the original image of grey-scale lightening as shown above. The value grey – scale for this image is from 0 to 255. In a single band of the electromagnetic spectrum (infrared, visible light, ultraviolet,

etc.), grey – scale images frequently result from measuring the intensity of light at each pixel. In a case like this when only a given frequency is captured they are monochromatic proper. They can also be created from full color image.

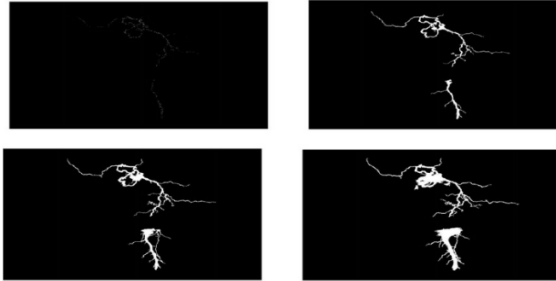


Fig.1.2 : Gray – Scale Lightening

Some segmented images of previous gray – scale original image using different values for segmentation are shown in Fig 1.2. This proves an image cannot have one segmentation that can be considered to be “accurate”. Only in the brain of the observer, an “accurate” segmentation exists, which can change not only among observers, but also within the same observer at different instants. In many line of work, like Face Recognition, Medical Imaging, Machine Vision, Fingerprint Recognition, Script Recognition etc. image segmentation is applied.

II. TEXTURE SEGMENTATION

For more than 50 years understanding of processes occurring in the early stages of visual perception has been a primary research topic. For regular properties like color, brightness, size and the slopes of lines composing figures pre – attentive segmentation occurs strongly. Research into the statistical properties of preattentively discriminable texture was started by Julesz in early 1960’s.

2.1 Definition

The regular repetition of an element or pattern on a surface is called as texture. Texture Segmentation, specifically it deals with texture segmentation of an image to find out the different types of textures present in the image to extract boundaries between major texture regions. A measurement of the variation of the intensity of a surface, quantifying properties such as regularity, smoothness and coarseness. You can also explain with term is color map. Texture is mapped onto an already available surface. In computer graphics there are deterministic (regular) and statistical (irregular) texture. It’s often used as a region descriptor in image analysis and computer vision. The three principal approaches used to describe texture are structural, spectral and statistical. Apart from the level of grey and color

texture is a spatial belief indicating what characterizes the visual homogeneity of given zone of an image in a infinite (true) image which generate another image based on the original texture and finally analyze these two fragments by classifying them in a different or a same category. In other words, we can also say that the main objective is to decide if texture samples belong to the same family by comparing them.

By using filter – bank model the process is bring to conclusion, dividing and decomposing of an input image into numerous output image is prepared by a set of linear image filters working in parallel which is used by the filter – bank model. These filters gives rise to concept of joint space / spatial – frequency decomposition by simultaneously concentrate on local spatial interactions and on particular range of frequencies.

2.2 Filters

In optics, device that let light pass on which have certain properties like particular range of wavelengths, i.e., range of colours of light, while blocking the rest. Mathematical operations carry out on an image represented as a sampled, discrete – time signal to enhance or reduce certain aspects of that signal in digital image processing is a Filter. Filtering is process which is used in Fourier transform for signal processing in frequency domain. Depending upon the relationship between input and output, filter may be linear or non – linear.

For the detection or removal of anomalous or unwanted frequencies from an input signal linear filters are used behaving according to the Gaussian statistical law. In high frequency components like small details and edges blurring is caused by linear filter.

Being derived from the neighboring pixels non – linear filters use estimates to detect anomaly. When applied for edge detection or spike detection non – linear filters normally provides better outcomes. Examples of filters are Long – pass, Polarizer, Band – pass, Median, Laplacian, Sobel, Short – pass, etc.

2.2.1 Different Types of Filters

- Low Pass: Frequencies lower than specified frequencies are allowed to pass.
- High Pass: Frequencies higher than specified frequency are allowed to pass.
- Band Pass: Frequencies in a limited range of are allowed to pass, it is a combination of low and high pass filters.
- Band Stop: Frequencies outside a limited range of frequencies are allowed to pass.

- All Pass: The phase relationship is altered in all – pass filter categorize of Band – pass filters.

III. DIFFERENT METHODS OF TEXTURE SEGMENTATION

The different methods for texture segmentation are :

- Edge detection
- Content – based image retrieval
- Thresholding
- Markov random fields
- Supervised segmentation
- Unsupervised segmentation
- Clustering methods
- Region – based
- Histogram – based and
- Gabor filter

All these techniques are backbone for texture segmentation also planned to design a novel approach for region detection and segmentation for the texture.

3.1 Gabor Filter

A Gabor filter is a linear filter used for edge detection in image processing which is named after Dennis Gabor. Gabor filter frequency and orientation representations are similar to those of human visual system, for texture representation and discrimination it has been found to be remarkably appropriate.

A sinusoidal plane wave has been modulating a 2D Gabor filter which is a Gaussian kernel function in the spatial domain. From one parent wavelet all filters can be generated by dilation and rotation, thus the Gabor filters are self-similar.

A harmonic function multiplied by a Gaussian function gives Gabor filter's impulse response. The convolution of the Fourier transform of the Gaussian function and the Fourier transform of the harmonic function is the Fourier transform of a Gabor filters impulse response because of Convolution theorem (the multiplication convolution property). Orthogonal directions are represented by an imaginary and a real component of the filter.

The two components may be shaped into a complex number or used individually.

Complex

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left\{-\frac{x^2 + \gamma^2 y^2}{2\sigma^2}\right\} \exp\left\{i\left(2\pi\frac{x'}{\lambda} + \psi\right)\right\}$$

Real

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left\{-\frac{x^2 + \gamma^2 y^2}{2\sigma^2}\right\} \cos\left(2\pi\frac{x'}{\lambda} + \psi\right)$$

Imaginary

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left\{-\frac{x^2 + \gamma^2 y^2}{2\sigma^2}\right\} \sin\left(2\pi\frac{x'}{\lambda} + \psi\right)$$

where,

λ is the wavelength of the sinusoidal factor,

θ is the direction of the normal to the parallel stripes of a Gabor function,

ψ is the phase offset,

σ is the sigma of the Gaussian envelope,

γ is the spatial aspect ratio and specifies the ellipticity of the cooperation of the Gabor function.

Colour has been shown to enhance intensity texture features as well as composing an intensity independent-pattern. Both RGB and HSL colour space are suitable for applying Gabor filters. Concerning the Gabor features, the phase energy supplements the amplitude energy, showing similar discrimination capabilities. Since the Fourier domain is used for filter bank design as well as its implementation, the Fourier transform is the main element of the Gabor transform.

In our paper, texture image segmentation using Gabor filters is implemented because of their good spatial and spatial frequency localization. These are of two types:

- Traditional Gabor Filter
- Circular Gabor Filter

IV. TRADITIONAL GABOR FILTERS

4.1 GEF's and Gabor Filter

Because of their good spatial and spatial frequency localization Gabor filters are extensively used for texture segmentation. Short synopsis of Gabor Elementary Functions (GEF's) and Gabor filters are given below. Daugman extended GEF to 2D but they were first defined by Dennis Gabor.

A GEF is given by

$$H(m,n) = g(\mathbf{m}', \mathbf{n}') * \exp[j2\pi(Pm + Qn)]$$

where,

$$(\mathbf{m}', \mathbf{n}') = (m \cos\theta + n \sin\theta - m \sin\theta + n \cos\theta)$$

$(\mathbf{m}', \mathbf{n}')$ represent rotated spatial domain rectilinear coordinates. Letting (p,q) denote frequency-domain rectilinear coordinated, (P,Q) represent a specific 2D

frequency. At frequency the complex exponential is a 2D complex sinusoid,

$$F = \sqrt{(P^2 + Q^2)}$$

and $\phi = \tan^{-1}(P/Q)$ specifies the orientation of the sinusoid. The function $g(m,n)$ is the 2D Gaussian.

$$g(m,n) = (1/\pi S_m S_n) \exp\{-1/2[(m/S_m)^2 + (n/S_n)^2]\}$$

where, S_m and S_n characterize the spatial extent and bandwidth of the filter. Therefore, the GEF is a Gaussian that is altered by a complex sinusoid. It is shown that the Fourier transform of $h(m,n)$ is

$$H(p, q) = \exp\{-1/2[(S_m[p - P])^2 + (S_n[q - Q])^2]\}$$

where,

$$[(p - P)', (q - Q)'] = [(p - P) \cos \theta + (q - Q) \sin \theta, -(p - P) \sin \theta + (q - Q) \cos \theta]$$

Therefore, from above 3 equations, the GEF's frequency retort has the shape of a Gaussian. The Gaussian's minor and major axis widths are determined by S_m and S_n , it is rotated by an angle with respect to the positive p - axis, and is centered about the frequency (P, Q) . Therefore, the GEF works as a band pass filter.

Letting $S_m = S_n = S$ is a reasonable design choice in maximum case. If it is assumed that $S_m = S_n = S$, then, the parameter θ is not required and the equation (1) of GEF simplifies to

$$h(m, n) = \left(\frac{1}{\pi \sigma^2}\right) \exp\left\{-\frac{1}{2}\left(\frac{m^2 + n^2}{\sigma^2}\right)\right\} \exp[j2\pi(Pm + Qn)]$$

We now define the Gabor filter by

$$t(m,n) = Oh(i(m,n)) = |i(m,n)| \times h(m,n)$$

where, i is the image and t is the output.

4.2 Filter Design

Find the Gabor filter that best distinguishes B and A in the output $t(m,n)$, given a bipartite textured image $i(m,n)$ consisting of known textures B and A. An output image $t(m,n)$ can be produced by a properly designed Gabor filter exhibiting some kind of discontinuity at the texture boundaries.

Gabor filter is a design which produces a step in $t(m,n)$ at texture boundaries when the two textures differ from each other. Case in which two textures differ is focused primarily. Therefore, the anticipated filter - design algorithm struggles to give a Gabor filter that produces a step in $t(m,n)$. The parameters P, Q and (1) and (5) are the ones by which Gabor filters are determined.

The Gabor filter is determined by the factors P, Q and as per (2.4) and (2.5), among the space of all possible Gabor filters. The space of all possible Gabor filters the best Gabor filter must be found, as determined by (P, Q, S) . To do this a metric for assessing filter quality is required. By the amplitude and slope of the step it produces in the case of a step signature the quality of a Gabor filter can be concluded.

4.2.1 Filter Design Algorithm

Filter Design Algorithm is based on two things:

- Based on the structure of the textures, select a value of S .
- Within each texture sample, to neighborhoods apply concurrently a huge number of Gabor filters (P, Q, S) about a set of randomly selected points. More exclusively, for each texture B and A:

➤ Within a texture select randomly a set z of points.

➤ Compute $[F_{M,N}(P, Q)]$ for each point $(M, N) \in z$

$$F_{M,N}(U, V) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} i(m, n) g(m - M, n - N) \times \exp[-j2\pi(Um + Vn)] dm dn$$

where, g is the Gaussian (2) having parameters $S_1 = S_2 = S$ and i is the given texture sample of i centered at (M, N) let the Windowed Fourier transform (WFT) as F and the window function as g .

Since (6) is implemented in exercise as an $X \times X$ DFT (Discrete Fourier Transform), $|F_{M,N}(P, Q)|$ tally to the output of a Gabor filter (P, Q, S) applied to point (M, N) , the calculation of FMN is equivalent to concurrently applying $X \times X$ different Gabor filters at the point (M, N) . Hence, This computation successfully extent or measure the range of filter center frequencies (P, Q) .

4.3 Algorithm Issues

4.3.1 Gabor - filter Application via WFT

The second step in our design method needs the parallel of a family of Gabor filters to an image at a point. This is a particular case of applying a Windowed Fourier Transform at the image given by the equation as below:

$$F_{M,N}(P, Q) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} i(m, n) w(m - M, n - N) \times \exp[-j2\pi(Pm + Qn)] dm dn$$

where, w is the window function, the image to be transformed is i , and a function of frequency is $f(P, Q)$ and window position (M, N) . To show this, let u be the result of convolution of an image $i(m,n)$ with a GEF $h(m,n)$:

$$u(m, n) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} i(m, n) \times h(m, n) = i(\alpha, \beta) h(m - \alpha, n - \beta) d\alpha d\beta$$

Expanding h and considering one specific point in the convolve (M,N) gives

$$u(M, N) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} i(\alpha, \beta) \times g(M - \alpha, N - \beta) \exp[j2\pi(P(M - \alpha) + Q(N - \beta))] d\alpha d\beta$$

After rearranging terms, and letting the window function $w(m,n) = g(-m,-n)$

$$u(M, N) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} i(\alpha, \beta) g(\alpha - M, \beta - N) \exp[-j2\pi(P\alpha + Q\beta)] d\alpha d\beta$$

On continuous function the development above is based on. The WFT is estimated as a DFT for the discrete case. Then (P,Q) refer to discrete special frequencies and (M,N) refer to image pixels. An image and truncated Gaussian centered are multiplied at pixel (M, N) for the DFT implementation. To the image is estimated by the DFT at pixel (M,N) the application of a family of Gabor filters, where center frequency of each filter is tally to frequencies of the DFT. Therefore, at a single point calculating a single discrete WFT is comparable to deciding the output from a family of Gabor filters, the frequency domain of the image is span by the center frequencies of the filters.

Obviously, the filter – selection algorithm does not consider all possible center frequencies. Experience shows that this is not necessary because step quality degrades gracefully with changes in Gabor – filter center frequency i.e., (Gabor filters having center frequencies (U +δU,V +δV) tend to perform nearly the same as filters having center frequency (U,V).

4.4 Filter Design Algorithm Discussion

Considering the two cases for H(m,n) which were discussed above, the following two algorithms were developed. As per the requirement, it was assumed that, a bipartite textured image I(m,n) was given and whose both corresponding textures B and A are known . The under consideration algorithm will implement the Gabor filter, which discriminates the texture B and A in the output image T(m,n). An output image t(m,n) would be produced by a perfectly designed Gabor filter which exhibits any kind of signature or else discontinuity at the boundary.

- When the two textures are differing from each other, then at the texture boundary the Gabor filter performs a step change in t(m,n).

- When there is a case, if the two textures B and A may have same value but have offset value different to each other.

Prior to that a case the correctly selected Gabor filter will create a valley of discontinuity at the corresponding texture boundary. This very form of output is referred as a valley signature.

The main objective is to focus on the case where the corresponding two textures differ. Hence the suggested filter design algorithm strives to provide the Gabor filters to produce step change in t(m,n). Mostly on theoretical basis a Gabor filter is determined by the parameters (P,Q,S). Being careful while choosing the values of these three parameters, this will design the optimal form of Gabor filter.

4.4.1 Algorithm – 1

- Step 1 : Select values for P,Q,S
- Step 2 : Then Converting the input image into a 2D matrix, say I(m,n)
- Step 3 : Then Calculating the value of H i.e. the impulse response of the filter,(which is also a 2D matrix) by the following formula given below:
 $H(m,n) = 1 / 2\pi S \exp\{-(x^2 + y^2) / 2 * s^2\} \{ \exp[j2\pi(Pm + Qn)] \}$
- Step 4 : Computation of the convolution of I(m,n) with H(m,n), say it is t(m,n), which is corresponding matrix to the output image. $t(m,n) = Oh(i(m,n)) = |I(m,n) \times H(m,n)|$ where, Oh represents the filter.
- Step 5 : Print the output image. If it cleanly discriminates the two textures good, stop. Otherwise go to Step 1 to select other values for P, Q, S & repeat the procedure.

4.4.2 Algorithm – 2

- Step 1 : Select values for U,V,Sm,Sn
- Step 2 : Then Converting the input image into a 2D matrix, say I(m,n)
- Step 3 : Then Calculating the value of H i.e. the impulse response of the filter,(which is also a 2D matrix) by the following formula given below:
 $H(x,y) = 1 / 2\pi S_m S_n \exp\{-(1/2)[(m/S_m)^2 + (n/S_n)^2]\} \{ \exp[j2\pi(Pm + Qn)] \}$
- Step 4 : Computation of the convolution of I(m,n) with H(m,n), say it is T(m,n), which is corresponding matrix to the output image.
 $T(m,n) = Oh(i(m,n)) = |I(m,n) \times H(m,n)|$ where, Oh represents the filter.
- Step 5 : Print the output image. If it cleanly discriminates the two textures good, stop. Otherwise go to Step 1 to select other values for P, Q, S & repeat the procedure.

V. CIRCULAR GABOR FILTER AND ROTATION INVARIANCE

5.1 Preview

Gabor filter is a powerful tool in texture analysis. Traditional Gabor function (TGF) represents a Gaussian function modulated with the help of an oriented complex sinusoidal signal. It is mathematically represented as

$$G(x,y) = g(x,y) \times \exp(2\pi i f(x \cos \theta + y \sin \theta))$$

Here $g(x,y) = (1/2\pi S^2) \times \exp(-(x^2 + y^2)/2S^2)$ (it is assumed that $g(x,y)$ is isotropic). The parameter f & θ represent the frequency and the orientation of the sinusoidal signal respectively. $g(x,y)$ is the Gaussian function with scale parameter S , f , θ and S constitute the parameter space of Gabor filters where θ lies in the interval $[\theta^\circ - 360^\circ]$. Gabor filters have many advantages over the Fourier transform. So it can get the best possible location in both the frequency and spatial domain.

5.2 Circular Gabor Filter & Rotation Invariance

The Gabor filters are very useful for the detection of texture direction. This is the main advantage of the traditional Gabor filters. Whereas when we consider rotation invariant texture analysis, then the orientation of texture becomes insignificant. Thus the traditional Gabor filters are very much less suitable for the specific purpose. The variation of sinusoidal grating of the TGF is in only one direction. If then the sinusoidal variation is in all orientations, it is known as circular symmetric.

These results in a new version of Gabor filter called Circular Gabor filter (CGF). The circular Gabor filter is defined as follows:

$$G(x, y) = g(x, y) \exp\{2\pi i F(\sqrt{x^2 + y^2})\}$$

where, F represents the central frequency for a circular Gabor filter. The properties of the circular Gabor filters can be more explicit in their frequency domain.

The Fourier representation of the circular Gabor filter is as follows:

$$Fourier(u, v) = \frac{\sqrt{(2\pi)}}{2} \alpha \exp\left\{-\frac{(\sqrt{u^2 + v^2} - F)^2}{2\alpha^2}\right\}$$

where, $\alpha = 1/\pi S$

In case of Gabor based texture, the analysis of the texture property of every single pixel is provided by the projection of the textured surface $I(x,y)$ onto a complex Gabor wavelet. That is (here it is used circular Gabor filter):

$$P = \int \int I(x, y) g(x, y) \exp\{i2\pi F\sqrt{x^2 + y^2}\} dx dy$$

Consider a rotation of texture image $I(x_0, y_0)$ by an amount of θ . The projection of $I(x_0, y_0)$ becomes

$$P' = \int \int I(x', y') g(x, y) \exp\{i2\pi F\sqrt{x^2 + y^2}\} dx dy$$

where,

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos \Delta\theta & \sin \Delta\theta \\ -\sin \Delta\theta & \cos \Delta\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

We have $dx \cdot dy = dx' \cdot dy'$ and $x^2 + y^2 = x'^2 + y'^2$

Thus the above equation can be represented as follows

$$P' = P = \int \int I(x', y') g(x', y') \exp\{i2\pi F\sqrt{x'^2 + y'^2}\} dx' dy'$$

From above equations it is obtained that $p = p_0$. Therefore it can be proved that, when we rotate an image, its very projection onto a circular Gabor wavelet never changes but remains the same. So this property provides the rotation invariance.

5.3 Selection of Parameters

When we consider circular Gabor filter, F and S should satisfy the condition with respect to bandwidth b as

$$F \times S = \lambda(2b + 1)/(2b - 1)$$

where, $B = \log_2 f_1/f_2$, $\lambda = \sqrt{(2 \ln 2)}/2\pi$ with f_1 and f_2 the frequencies corresponding to the half peak magnitude of circular Gabor Fourier response.

For a texture image of size $N * N$, the frequency F is often selected as follows:

$$\sqrt{2} \{1, 2, 4, 8, 16, N/4\}/N$$

This choice implies that finer frequency resolutions are in lower frequencies than in higher frequencies, i.e. it emphasizes the lower frequencies. This choice may result in very good texture representation. However may be the good reconstruction it does not proved to be a good segmentation. When we consider lower frequencies they may provide good texture segmentation results. Frequency can be chosen from the interval of $[-0.5-0.5]$ by an image of size N . The

lowest and the highest frequency being FH and FL respectively.

$$FH = 0.25 + 2(I - 0.5)/N \quad 0.25 \leq FH < 0.5$$

$$FL = 0.25 + 2(I - 0.5)/N \quad 0 < FL < 0.5$$

5.4 Texture Segmentation

When it comes to recognizing the textures, the texture features were calculated from the filtered image as per the following equation

$$\Psi(x,y) = r2(x,y) \otimes m(x,y)$$

where, r2(x,y) represents how much is the energy of the filtered image and \otimes shows the convolution . m(x,y) represents the mask to obtain the texture segments. Finer texture measurements can be detected by mask windows of minute size. For the local texture energy a Gaussian window is being used. We require Pattern clustering algorithm for identifying the regions.

5.5 Invariant Texture Segmentation Using Circular Gabor filter

5.5.1 Proposed Algorithm

Step 1 : Selecting the value of F.

Step 2 : Then Converting the input image into a 2D matrix respectively.

Step 3 : Computing the impulse response of the filter by using the following formula

$$G(x,y) = g(x,y) \times \exp\{2\pi i f(x \cos \theta + y \sin \theta)\}$$

$$g(x,y) = (1/2\pi S^2) \times \exp\{-(x^2 + y^2)/2S^2\}$$

Step 4 : Then the Filtering the input image by passing it through the filter whose impulse response is as calculated in Step 3, say r(x,y) is the filtered output.

Step 5 : Compute the energy E of the filtered image as follows E = r2(x,y)

Step 6 : Calculate the segmented image from the corresponding filtered image as follows.

$$\phi(x,y) = E \otimes m(x,y),$$

where m(x,y) is the mask window. The size of the window was calculated by its standard deviation Ss, we choose Ss = S / sigma

$$\text{So, } m(x,y) = (1/8\pi\sigma^2) \times \exp\{-((n1^2 + n2^2)/4\sigma^2)\}$$

Step 7 : Print the output image $\phi(x,y)$

5.6 Assessments

- When we compare with the traditional Gabor filters, the circular Gabor filters certainly produces better results for texture segmentation.
- We found that the circular Gabor filter is kind of rotation invariant, whereas the traditional Gabor filters are not, and this was proved mathematically.
- When it comes to the traditional selection scheme, two frequencies are chosen. But in the case of central frequency scheme for the circular Gabor filter produces better segmentation results as compared to the other.

VI. IMPLEMENTATION RESULTS

6.1 Algorithm of Traditional Gabor Filter

Step 1 : Select values for P,Q,S

Step 2 : Then Converting the input image into a 2D matrix, say I(m,n)

Step 3 : Then Calculating the value of H i.e. the impulse response of the filter, (which is also a 2D matrix) by the following formula given below:

$$H(m,n) = 1 / 2\pi S \exp\{-(x^2 + y^2)/2*s^2\} \{ \exp[j2\pi(Pm + Qn)] \}$$

Step 4 : Computation of the convolution of I(m,n) with H(m,n), say it is t(m,n), which is corresponding matrix to the output image.

$$t(m,n) = Oh(i(m,n)) = |I(m,n) \times H(m,n)|$$

where, Oh represents the filter.

Step 5 : Print the output image. If it cleanly discriminates the two textures good, stop. Otherwise go to Step 1 to select other values for P, Q, S & repeat the procedure.

6.2 Results for Traditional Gabor Filter

The input image for texture segmentation is shown in Fig 6.1. Now, Select the values of P,Q and Theta to get the Perfect output. If it is not good, again change the values and check for the better output.



Fig.6.1: Original Image

The colour image is now converted into Grey – Scale image. So, it will be easy to perform further operations on the input image.

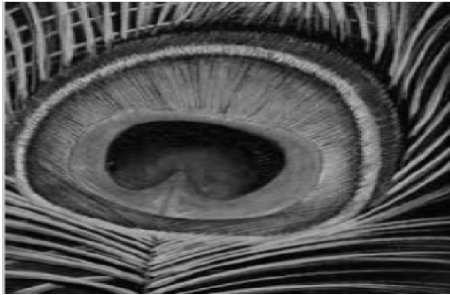


Fig.6.2: Grey – Scale image of the input image

When $P = 0.8$, $Q = 1.0$ and $\Theta = 30$ degrees, the segmented image appears as shown in Fig 6.2. The edges of the grey scale image texture are highlighted depending on the values of the P and Q values.

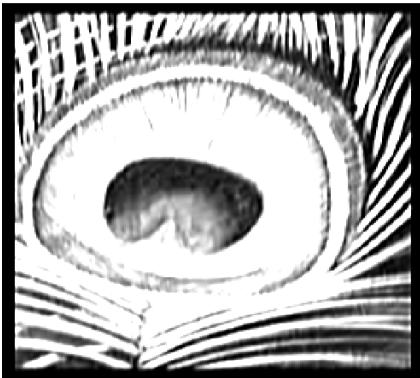


Fig.6.3: Output when $P = 0.8$, $Q = 1.0$, $\theta = 30$

When $P=1.1, Q=0.9$, and $\Theta=0.5$, we can see that the image edges are more clear as shown in Fig 6.4.

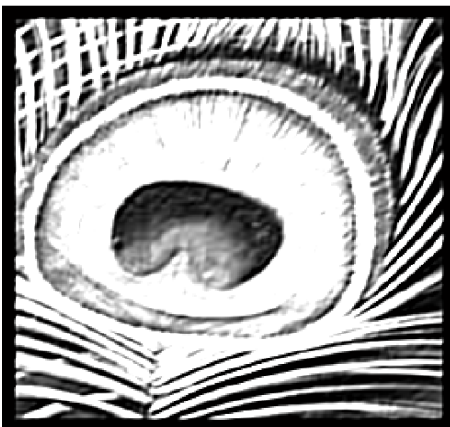


Fig.6.4 : Output when $P = 1.1$, $Q = 0.9$, $\theta = 0.5$

When $P = 0.8$, $Q = 0.9$ and $\Theta = 0$, the output Fig 6.5 is as much good as before values. But there is not Perfect Boundaries between two different textures



Fig.6.5: Output when $P = 0.8$, $Q = 0.9$, $\theta = 0$

When $P = 0.8$, $Q = 1$ and $\theta = 0$, the output of texture segmentation appears as Fig 6.6.

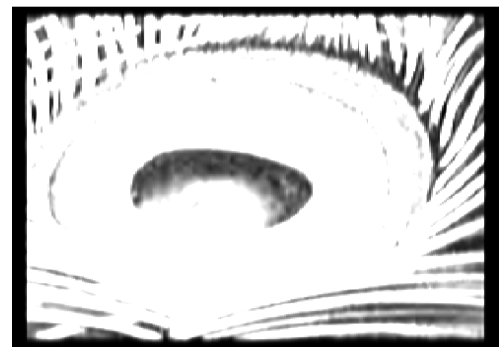


Fig.6.6: Output when $P = 0.8$, $Q = 1$, $\theta = 0$

When $P = 0.8$, $Q = 0.9$, $\theta = 30$, the output is darker than before outputs because of the value of θ and P , Q as shown in Fig 6.7.

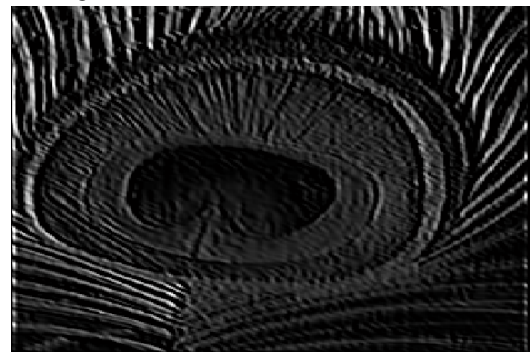


Fig.6.7: $P = 0.8$, $Q = 0.9$, $\theta = 30$

Similarly, if the values are slightly modified to $P = 0.9$, $Q = 1.2$, $\theta = 0.25$ then the output has much brighter than Fig 6.7 that we can see in Fig 6.8.

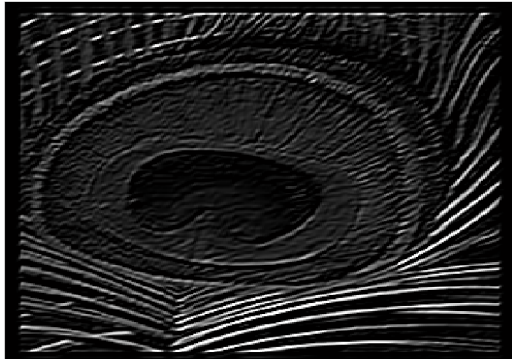


Fig.6.8: $P = 0.9$, $Q = 1.2$, $\theta = 0.25$

6.3 Algorithm of Circular Gabor Filter

Step 1 : Read the input image and then Converting the input image into a 2D matrix, say $I(m; n)$.

Step 2 : Select values for F , σ & λ .

Step 3: Parameters should Satisfy

$$F\sigma = \lambda \frac{(2^B + 1)}{(2^B - 1)}$$

Step 4: Response of Circular Gabor Filter

$$G(x, y) = g(x, y) \exp(2\pi F(\sqrt{x^2 + y^2}))$$

$$g(x, y) = \frac{1}{2\pi\sigma^2} \exp(-(x^2 + y^2)/2\sigma^2)$$

The parameter f represent the frequency. $g(x, y)$ is the Gaussian function where θ lies in the interval $[0, 360]$.

Step 5 : convolute input image and filter function $G(x,y)$.

Step 6: Print the output image.

6.4 Results for Circular Gabor Filter

The input image is shown in the Fig 6.9 which is a RGB image.



Fig.6.9: Original Image

The colour image is now converted into Grey – Scale image. So, it will be easy to perform further operations on the input image.

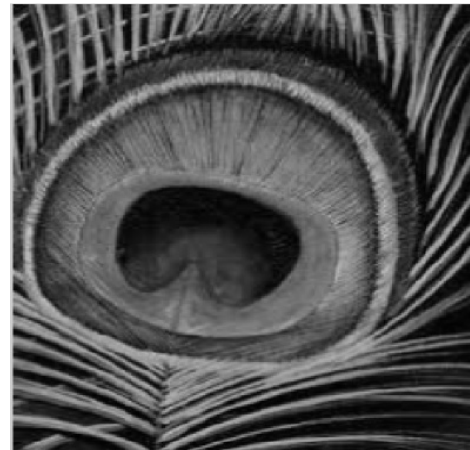


Fig.6.10: Grey – Scale image of the input image

Now, select the values of F and λ so that it must satisfy the condition mentioned in the circular algorithm. θ value may be anything since it is a rotation invariant filter.

When, $F = 0.8$, $\lambda = 0.4$ and $\theta = 0$, the texture segmentation of the image is shown in Fig 6.8.

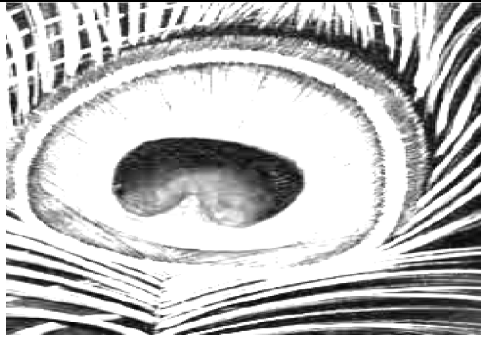


Fig.6.11: $F = 0.8$, $\lambda = 0.4$, $\theta = 0$

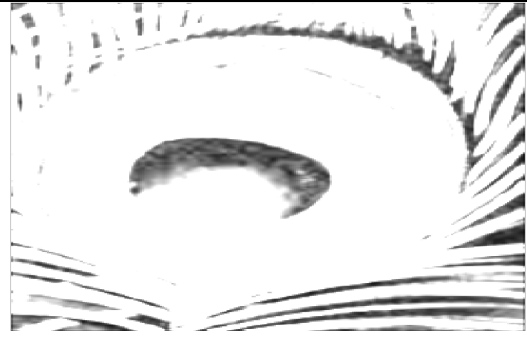


Fig.6.14: $F = 1.4$, $\lambda = 0.8$, $\theta = 0$

Now, change the values of F, Lambda and theta to 1.2, 0.3, 30. So the obtained image is Fig 6.12.

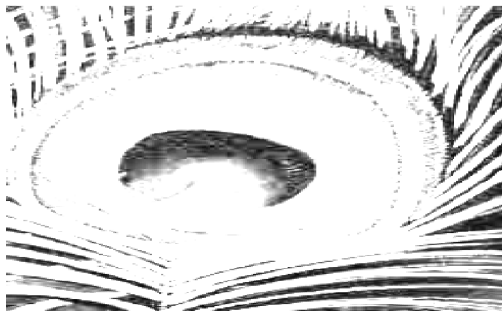


Fig.6.12: $F = 1.2$, $\lambda = 0.3$, $\theta = 30$

When $F=0.7$, $\lambda=0.3$ and $\theta=0$, only the borders are highlighted depending on texture shown in Fig 6.13.

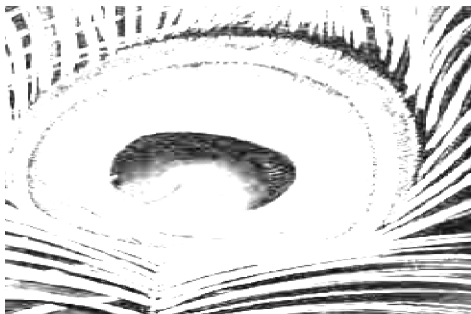


Fig.6.13: $F = 0.7$, $\lambda = 0.3$, $\theta = 0$

When $F = 1.4$, $\lambda = 0.8$, $\theta = 0$, the output of segmentation is as Fig 6.14.

The texture segmentation of input image is as Fig 6.15, when $F = 1.7$, $\lambda = 0.3$ and $\theta = 60$.

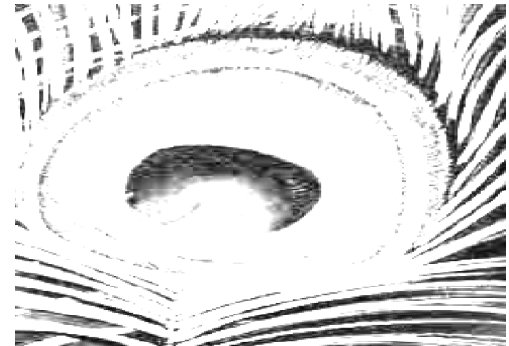


Fig.6.15: $F = 1.7$, $\lambda = 0.3$, $\theta = 60$

When $F = 1.6$, $\lambda = 1$ and $\theta = 30$ the output is as Fig 6.16.



Fig.6.16: $F = 1.6$, $\lambda = 1$, $\theta = 30$

If $F = 1$, $\lambda = 0.2$ and $\theta = 0$, the result appears as Fig 6.17.



Fig.6.17: $F = 1$, $\lambda = 0.2$, $\theta = 0$

VII. CONCLUSION

An essential step in analysis applications and pictorial pattern is Image segmentation. Accuracy the failure or success of analysis procedures is determined by segmentation. Texture segmentation is established on partitioning an image into different regions of similar texture based on particular criteria.

This scheme demonstrated good performance in Traditional & Rotation invariant (Circular) texture segmentation. The projection of images onto circular Gabor filters is rotation invariant in nature. Texture Segmentation is an essential step in analysis applications and pictorial Pattern.

Our experiments show that this scheme demonstrates good performance in Traditional& rotation invariant texture segmentation.

Further research focus may be on testing the efficiency of this method on noisy images. Moreover, how to reduce the number of assumptions considered while selecting the filter parameters without compromising the segmentation results is an interesting problem to be investigated.

REFERENCES

- [1] Invariant Texture Segmentation Via Circular Gabor Filters,Jainguo Zhang, Tieniu Tan, Li Ma,National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences,Beijing 100080, P.R. China.
- [2] Texture Segmentation Using Optimal Gabor Filter by Dipesh Kumar Solanki, Khagswar Bhoi, National Institute of Technology Rourkela,Orissa, India.
- [3] Texture Segmentation: Different Methods ,Vaijinath V. Bhosle, Vrushsen P. Pawar, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-3, Issue-5, November 2013.