TEXTURE BASED SEGMENTATION

Submitted By-

Sanjeev Kumar(Roll no-204102309)

Ashlesha Medhi (Roll no-204102002)

Texture Based Segmentation

Motivation:-

The ability of the human observers to discriminate between textures is related to the contrast between key structural elements and their repeating patterns.As the natural world abounds with textured surfaces,so any realistic vision system that is expected to work successfully,must be able to handle such type of input.The Texture segmentation has been an important task in image processing. Basically, it aims at segmenting a textured image into several regions having the similar patterns.So an effective and efficient texture segmentation method will be very useful in applications like the analysis of aerial images,biomedical images and seismic images as well as the automation of industrial applications. Like other segmentation problems,the segmentation of textures requires the choice of proper texture-specific features with good discriminative power.

Introduction:-

Texture segmentation is the process of identifying regions with similar texture and separating regions with different texture.It is one of the early steps towards identifying surfaces and objects.The study of texture analysis has been done for a long time using various approaches.The texture feature extraction methods can be classified into three major categories, namely, statistical, structural and spectral. In case of statistical approaches, texture statistics such as the moments of the gray-level histogram, or statistics based on graylevel co-occurrence matrix are computed to discriminate different textures. In structural approaches, “texture primitive”, the basic element of texture, is used to form more complex texture patterns by applying grammar rules, which specify how to generate texture patterns. Finally, in spectral approaches, the textured image is transformed into frequency domain. Then, the extraction of texture features can be done by analyzing the power spectrum.

In this project, we propose a method of obtaining texture features directly from the gray-level image by computing the moments of the image in local regions. At first, the entropy of the binary image,thresholding of the image,the moments of a two-dimensional function, the computation of texture features from the moments, and an algorithm that uses these features to segment the texture images has been described,which is followed by its experimental results, along with its concluding remarks.

Texture Segmentation:-

The texture segmentation algorithm is given in the flowchart of Figure 1(a) and it consists of the following steps:

(1)Converting the RGB image into binary image

(2)Computing the entropy of the image

(3)Thresholding the image into two regions

Since the above algorithm is applicable only for those images that contain only one type of texture,so in the project we have segmented the different textured regions by computing the moments of the image.The flowchart of this segmentation algorithm is shown in Figure 1(b) .The algorithm of the texture segmentation is as follows:-

(1) Pre-processing is done so as to normalize the pixel range of the image in between [-1,1]

(2) Computing the image moments within a small window around each pixel,

(3) Performing an unsupervised clustering of a randomly selected points in the image, and

(4) Classifying every pixel in the image according to the results of step (3).

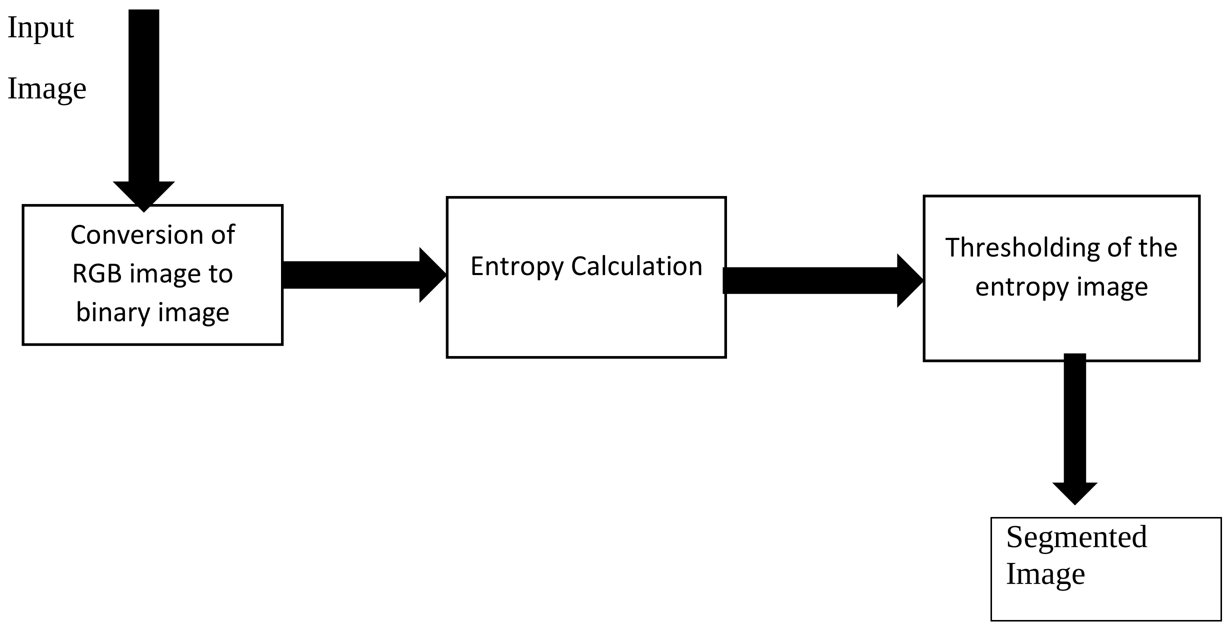


FIGURE 1(a). The flowchart of the texture based segmentation algorithm(by computing the entropy of the image)

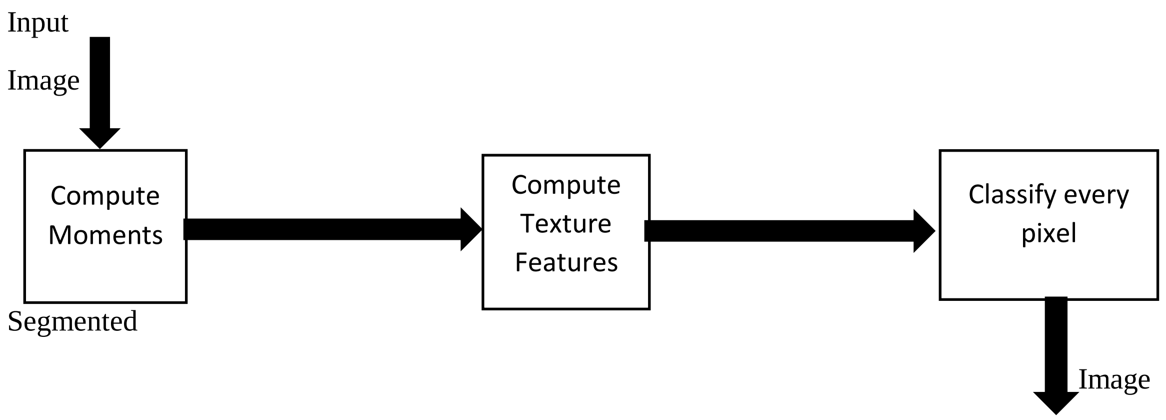


FIGURE 1(b). The flowchart of the texture based segmentation algorithm(by computing the moments of the image)

(a) Entropy:-

Entropy is the statistical measure of randomness of a binary image that can be used to characterize the image on the basis of its textures.

The entropy of an image is given below:-

$$H=-\sum_k p_k \log_2(p_k),$$

where ,K denotes the number of gray levels and p_k denotes the probability associated with gray level k.

In the project,we have first converted the RGB image into a binary image and then calculated the entropy of the image with the help of inbuilt function of the software.

(b)Thresholding:-

Thresholding is the simplest method of segmenting an image into two regions in digital image processing. The simplest thresholding methods replace each pixel of the image with a black pixel if the image intensity $I_{i,j}$ is less than some fixed constant T (that is, $I_{i,j}<T$),or a white pixel if the image intensity is greater than that constant.

In the project,we have thresholded the entropy image into black and white regions with the help of in-built function of the software.

(c)Moments:-

Our algorithm uses the moments of an image to compute texture features. The (p+q) th order moments of a function of two variables f(x,y) with respect to the origin (0,0) are defined as

$$pq=\int_{-\infty}^{\infty} f(x,y)x^{p}y^{q}dxdy \quad (A)$$

where p+q=0,1,2………The infinite set of moments {m_{pq},p+q=0,1,…..} uniquely determine f(x,y) and vice versa by the following equation

$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp\left(-j\,2\pi\left(ux+ vy\right)\right)\left[\sum_{p=0}^{\infty}\sum_{q=0}^{\infty} m_{p,q}\frac{(j\,2\pi)^{p+q}}{p!q!}u^{p}\, v^{q}\right]dudv$$

The moments are usually computed over some bounded region R. If the function is equal to one within the region and zero outside the region, the lower order moments (i.e.small values of p and q) have well defined geometric interpretations.Suppose if the moment m_{00} is the area of the region,then the moments m_{10} / m_{00} and m_{01} / m_{00} give the x and y coordinates of the centroid for the region, respectively.The moments m_{20}, m_{11}, and m_{02} can be used to derive the amount of elongation of the region, and the orientation of its major axis. The higher order moments give even more detailed shape characteristics of the polygons such as symmetry etc.

In this project, we have regarded the intensity image as a function of two variables, f(x,y).After that,we compute a fixed number of the lower order moments for each pixel in the image (here we have used $p+q \leq 2$). The moments of the image are computed within small local windows around each pixel. Given a window size, the coordinates are normalized to the range [-1,1], the pixel being at the origin. The moments are then computed with respect to this normalized coordinate system. This permits us to compare the set of moments computed for each pixel. Let W be the window width. We always choose the window width W to be odd so that the given pixel (i,j) is centered on a grid point. Let (i,j) be the pixel coordinates for which the moments are computed. For a pixel with coordinates (m,n) which falls within the window, the normalized coordinates (x_m,y_n) are given by

$$x_{m}=\frac{m-i}{W/2}\ ,y_{m}=\frac{n-j}{W/2}$$

Then the moments within a window centered at pixel (i,j) are computed by a discrete sum approximation of Equation (A) that uses the normalized coordinates (x_m, y_n)

$$pq=\sum_{-W/2}^{W/2}\sum_{-W/2}^{W/2} f(m,n)x^{p}y^{q}dxdy$$

This discrete computation of the set of moments for a given pixel over a finite rectangular window corresponds to a neighborhood operation, and, therefore, it can be interpreted as a convolution of the image with a mask.
The masks corresponding to the moments up to order two with a window size of three is shown below:-

m_{00}

1	1	1
1	1	1
1	1	1

$m_{10}m_{01}$

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

m_{20}

1	1	1
0	0	0
1	1	1

m_{11}

1	0	-1
0	0	0
-1	0	1

m_{02}

1	0	1
1	0	1
1	0	1

The masks shown above can be interpreted as local feature detectors. For example, the mask for m_{00} corresponds to a box averaging window and thus can be interpreted as computing the total energy within that box. The masks for m_{10} and m_{01} have the form of edge detectors or contrast detectors. They would respond to sudden intensity changes in the x and y directions, respectively. The second order moments are not as easy to interpret; the only exception being m_{11} which acts like a cross detector.

The window size should also be taken into consideration.For a larger window size, more global features are detected. This suggests that the choice of window size could possibly be tied to the contents of the image. The images with larger texture tokens would require larger window sizes whereas finer textures would require smaller windows.

The set of values for each moment over the entire image can be regarded as a new feature image. Suppose $M_k$denotes the kth such image. If we use n moments, then there will be n such moment images. In our project, we have used up to second order moments. That is, we have used m_{00}, m_{10}, m_{01}, m_{20}, m_{11}, and m_{02} which result in the images M_1 , M_2 , M_3 , M_4 , M_5 , and M_6 respectively.

These moments images are then passed through a Non-linear function of g(x) = tanhx to increase the separability.

(d)Clustering of the image:-

The c
regio
algor

1. Choosing the number of clusters K in the image,
2. Select at random K points from the data as centroids
3. Assign each data point to the closest centroid , that forms K clusters.
4. Compute and place the new centroid of each cluster.
5. Reassign each data point to the new closest centroid. If any reassignment took place, go to step 4, otherwise, the model is ready

Texture based segmentation Algorithm:-

By calculation of Entropy of the binary image:-

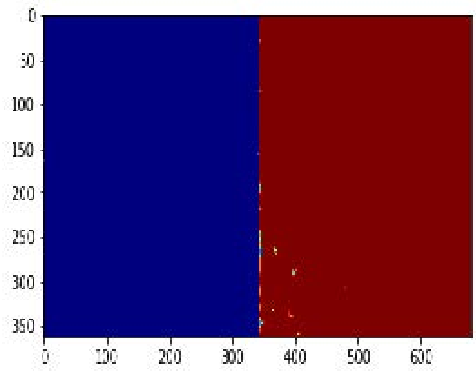
- An image of size 256x256 is taken.
- Convert the RGB image into a binary image
- Calculation of the entropy of the binary image
- Thresholding of the image into two distinct regions.

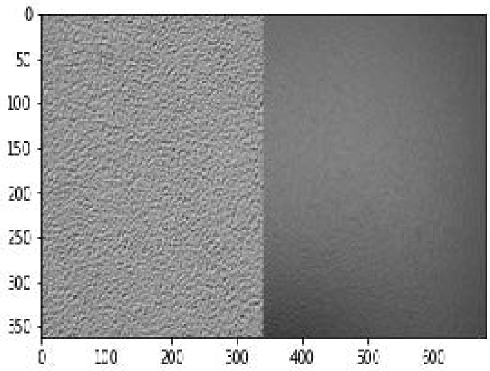
By calculation of Moment of the image:-

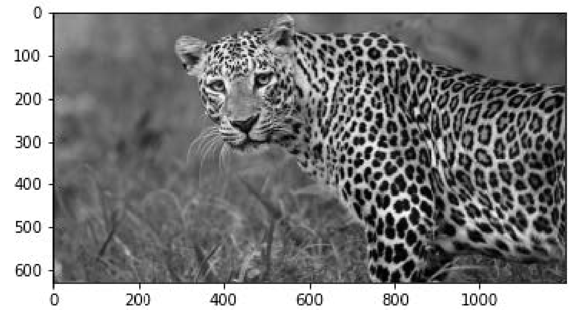
- An image of size 256x256 is taken.
- Pre-processing is done so as to normalize the pixel range of the image in between [-1,1]
- Taking a window size of 5x5,the moment of the entire image is calculated in order to extract some particular characteristics of the image.
- After that,the K-mean clustering of the image is done so as to segment the image on the basis of its textures.

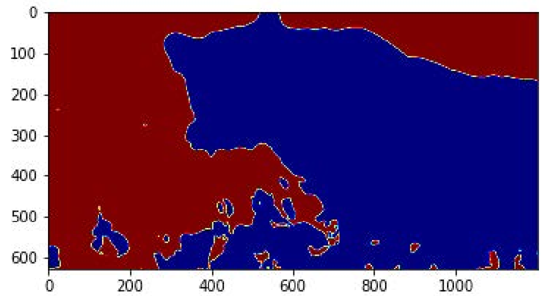
Segmentation Results

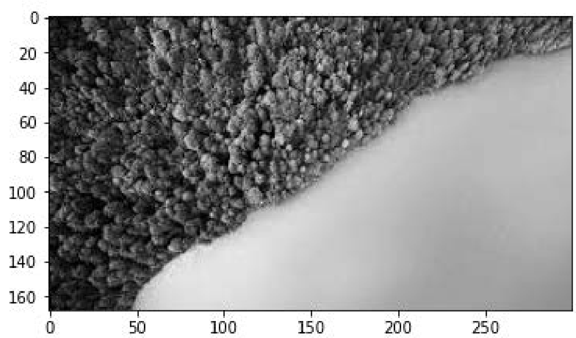
Entropy Based Method:

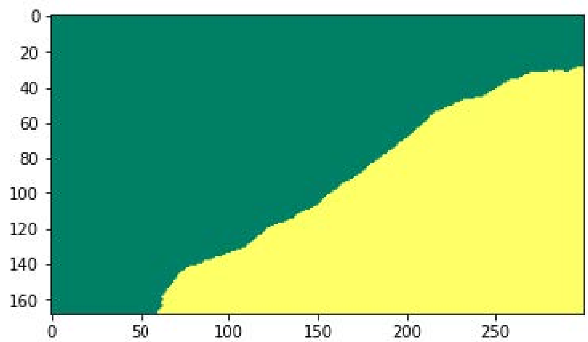




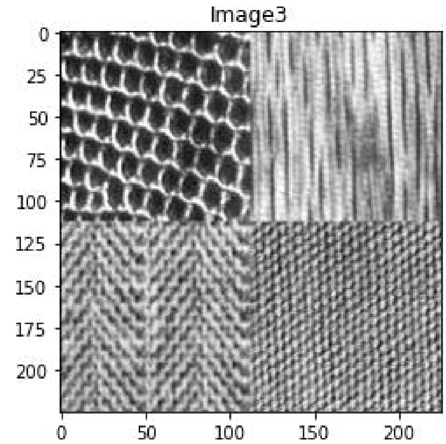


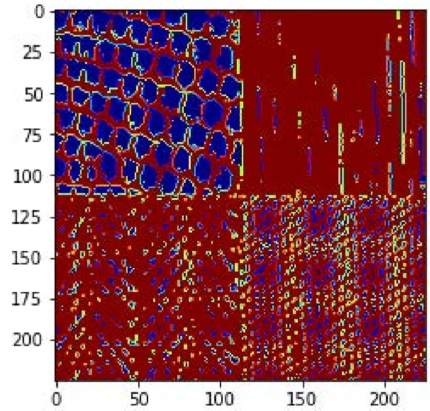


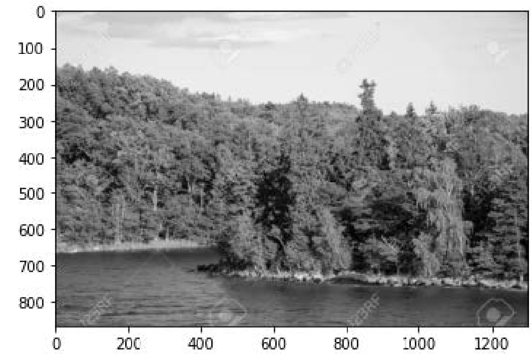


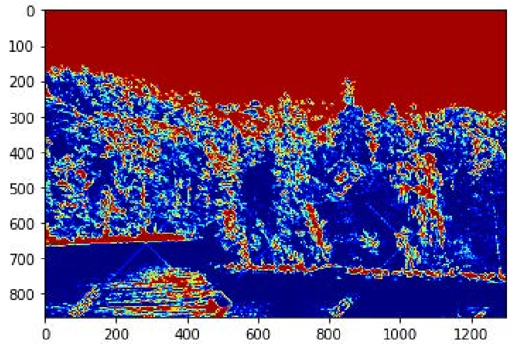


Moment-Based Method:









Conclusion :

The method used here is entropy based and moment based which are very simple method and doesn't require very huge domain knowledge ,The segmentation algorithm included with the moment model produces a efficient texture based algorithm. The model might not work on heavy clusters where the window size might need to be changed according to the image. Another drawback with the algorithm is that entropy is just one of the parameter that reflects texture difference. There may be as well others like GLCM based properties have turned out the most effective ones, Once these features are loaded these can be given to the ConvNets to finally cluster the similar pixels and hence get the desired results.

References :

- Texture Segmentation using Gabor Filters
  Naotoshi Seo, sonots@umd.edu

- Moment Based Texture Segmentation
  Mihran Tuceryan *European Computer-Industry Research Centre (ECRC) Arabellastraße 17 8000 München 81, Germany*

- K. I. Laws, *Textured Image Segmentation*. Ph.D. Thesis, University of Southern California, 1980.