World Journal of Engineering Research and Technology
WJERT
www.wjert.org
SJIF Impact Factor: 4.326

EDGE DETECTION FOR FOOTPRINT USING DISCRETE WAVELET TRANSFORM

[1]Riya Porel, [1]Sharanya Mazumdar, [1]Trishita Mukherjee, [1]Varun Kumar Sah, [2]Prof. Subhojit Malik and [2]*Prof. Samir Kumar Bandyopadhyay

[1]Students of E.C.E. 4th Year from Hooghly Engineering and Technology College.
[2]E.C.E. Department, Hooghly Engineering and Technology College.

Article Received on 04/04/2017 **Article Revised on 26/04/2017** **Article Accepted on 17/05/2017**

***Corresponding Author**
Prof. Samir Kumar Bandyopadhyay
E.C.E. Department,
Hooghly Engineering and Technology College.

ABSTRACT

Image processing and analysis based on the Continuous or Discrete Image Transforms are classic techniques. Amongst these, Discrete Image Transform is widely used in image filtering, data description, etc. Also, the Wavelet Theorem makes up very popular methods of image processing, de-noising and compression. Considering that the Haar functions are the simplest wavelets, these forms are used in many methods of Discrete Image Transforms and processing. In this article, it has been shown how the proper implementation of Discrete Wavelet Transform using Haar functions can be beneficial in extracting the edges of a processed sample footprint thus required in a Footprint Analysis Program for crime detection.

KEYWORDS: Discrete Wavelet Transform, Haar functions, Footprint Analysis.

INTRODUCTION

Image processing and analysis based on the Continuous or Discrete Image Transforms are classic techniques. Amongst these, Discrete Image Transform is widely used in image filtering, data description, etc. Also, the Wavelet Theorem makes up very popular methods of image processing, de-noising and compression. Considering that the Haar functions are the simplest wavelets, these forms are used in many methods of Discrete Image Transforms and processing.

In processing techniques using Efficient Wavelet Implementation, the transform operation is a linear transform that aims to reduce the entropy of the image. This operation is reversible and does not cause any loss of information to the image. An example of such a transform operation is Discrete Fourier transform (DFT), Discrete Cosine transform (DCT) or Discrete Wavelet Transform (DWT).

Disadvantage of this operation, using traditional (DFT, DCT, DST etc) transform is that it has only frequency resolution and no time resolution which implies that it is not possible to determine the time duration when all the frequencies are present in a signal. The inverse of traditional transforms provides us with the time resolution and not the frequency resolution.

The most recent and desirable solution to overcome the shortcoming of traditional transforms is the wavelet transform. The wavelet allows us to trade off the time and frequency resolution in different ways. Moreover, one of the wavelet transform advantages concerning data compression is that it tends to compact the input signal energy into a relatively small number of wavelet coefficients. Lowest resolution band coefficients have high energy and most of high frequency band coefficients are zero or have low energy, thus making the de-noising process relatively easier.

REVIEW WORKS

Classical methods of edge detection involve convolving the image with an operator (a 2-D filter),[6] which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions. There are an extremely large number of classical edge detection operators like Roberts, Sobel, Prewitt, Laplacian operators and canny.[7] Operators can be optimized to look for horizontal, vertical or diagonal edges. Classical Edge detectors usually fail to handle images with dull object outline or noisy images, since both the noise and the edges contain high frequency content. Attempts to reduce the noise result in blurred and distorted edges.[2][6] To reduce the influence of noise, many techniques have been developed. Filtering of images can be done with the Gaussian before edge detection. Methods have also been proposed to approximate the image with a smooth function.[8] Where the classical edge detectors fail to detect edges in noisy images, the proposed edge detection method works efficiently on images corrupted by noise and is able to differentiate between noise and real edges, thus detecting the actual edges.

Edge detection refers to the process of identifying and locating sharp discontinuities in an image.[1] The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. These methods take binary image as input and returns a binary image as output with 1's at pixels which are detected as edges, 0's at other pixels, which are not detected as edges. Since the abrupt change in brightness level indicates edge.[3,4] The transition in intensity in gray-scale image is relatively smooth in nature rather than abrupt as in case of segmented or binary images. The nature of intensity variation points to the application of derivative operators for detecting edges,[5] Application of derivative operators on intensity image produces another image, usually called gradient image as it reveals the rate of intensity variation.

Utility of Discrete Wavelet Transform in Edge Detection

In the Footprint Analysis Program for crime detection, sample footprints are collected (ie, efficiently photographed or taken impression of) from the crime scene. They are needed to be processed before developing a database with the samples for further comparisons with the database of samples collected from the most probable suspects related to the crime. The photographed images are firstly scanned, converted from RGB to Grayscale format, then to Binary and hence processed for Edge detection.

Edge detection refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. Classical methods of edge detection involve convolving the image with an operator (a 2-D filter), which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions. Canny Edge Detection technique is a popular method for this purpose.

But detection becomes difficult in noisy images, since both the noise and the edges contain high-frequency content. Attempts to reduce the noise result in blurred and distorted edges. Operators used on noisy images are typically larger in scope, so they can average enough data to discount localized noisy pixels. This results in less accurate localization of the detected edges.

Not all edges involve a step change in intensity. Effects such as refraction or poor focus can result in objects with boundaries defined by a gradual change in intensity. The operator needs to be chosen to be responsive to such a gradual change in those cases. Newer wavelet-based transformation techniques actually characterize the nature of the transition for each edge in order to distinguish accurately. Hence Discrete Wavelet Transform method has been preferably used for proper noise-free edge detection of the sample footprints and Haar functions being the simplest ones have been used.

General working structure of Discrete Wavelet Transform

In Discrete Wavelet Transform, an image signal can be analysed by passing it through an analysis filter bank followed by decimation operation. The analysis filter bank consists of a low-pass and high-pass filter at each decomposition stage. When the signal passes through these filters, it splits into two bands. The low-pass filter which corresponds to a differencing operation, extracts the coarse information of the signal. The high-pass filter, which corresponds to a differencing operation, extracts the detail information of the signal. The output of the filtering operation is then decimated by two. A two-dimensional transform is accomplished by performing two separate one-dimensional transforms. First, the image is filtered along the row and decimated by two, it is then followed by filtering the sub-image along the column and decimated by two. This operation splits the image into four bands namely, LL, LH, HL and HH respectively as shown in Fig.1

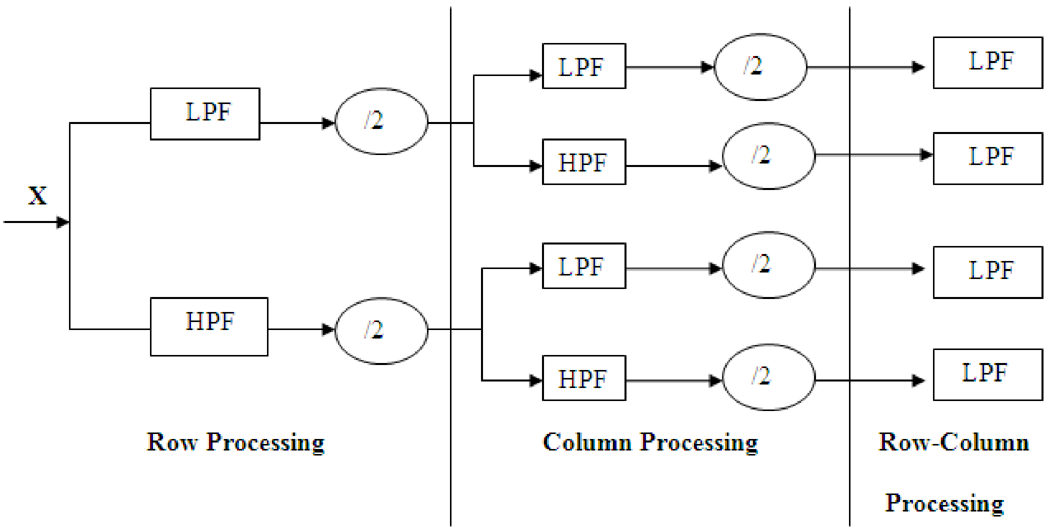

Fig 1: Wavelet Decomposition.

The 1st level of decomposition is shown as above. If Fig.2 is taken as the input image X (m, n), its 1st level of wavelet decomposition is given in Fig.3

Fig 2: Original Image.

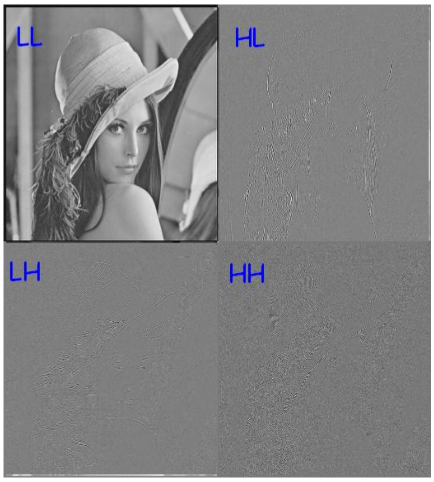

Fig 3: Image after Wavelet Decomposition.

Further decompositions can be achieved by acting upon the LL sub-band successfully and the resultant image is split into multiple bands as shown in Fig.4. The size of the input image and the size of the image at different levels of decomposition are illustrated in the same figure.

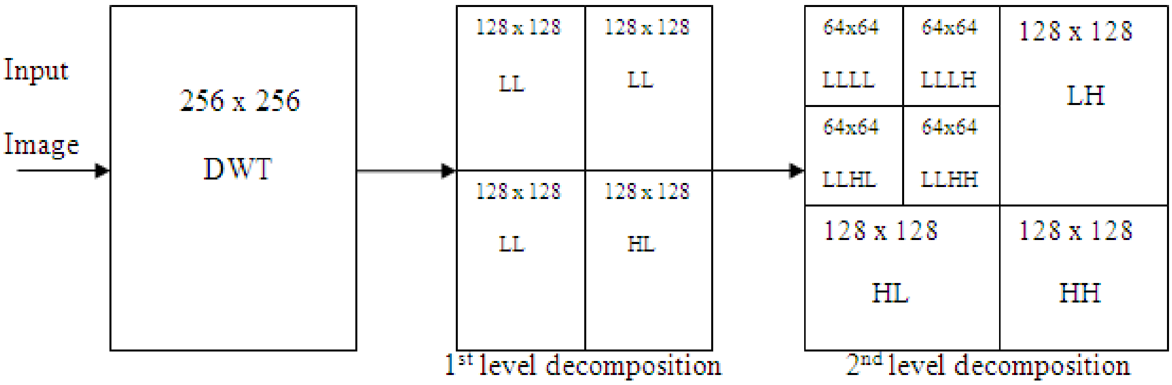

Fig 4: Second-level decomposition of the input image.

Haar Wavelets

The first DWT was invented by Hungarian mathematician Alfréd Haar. For an input represented by a list of numbers, the Haar wavelet transform may be considered to pair up input values, storing the difference and passing the sum. This process is repeated recursively, pairing up the sums to prove the next scale, which leads to differences and a final sum. In mathematics, the Haar wavelet is a sequence of rescaled "square-shaped" functions which together form a wavelet family or basis.

Haar wavelet compression is an efficient way to perform both lossless and lossy image compression. It relies on averaging and differencing values in an image matrix to produce a matrix which is sparse or nearly sparse. A sparse matrix is a matrix in which a large portion of its entries are 0. A sparse matrix can be stored in an efficient manner, leading to smaller file sizes. Here we will concentrate on grayscale images, however, RGB images can be handled by compressing each of the color layers separately. The basic method is to start with an image A, which can be regarded as an m × n matrix with values 0 to 255. In MATLAB, this would be a matrix with unsigned 8-bit integer values. We then subdivide this image into 8 × 8 blocks, padding as necessary. It is these 8 × 8 blocks that we work with.

Below is a 512 × 512 pixel grayscale image of the flying buttresses of the Notre Dame Cathedral in Paris:

Fig 5: Notre Dame Cathedral in Paris.

The following is the upper left 8 × 8 section of our image:

$$A = \begin{pmatrix} 88 & 88 & 89 & 90 & 92 & 94 & 96 & 97 \\ 90 & 90 & 91 & 92 & 93 & 95 & 97 & 97 \\ 92 & 92 & 93 & 94 & 95 & 96 & 97 & 97 \\ 93 & 93 & 94 & 95 & 96 & 96 & 96 & 96 \\ 92 & 93 & 95 & 96 & 96 & 96 & 96 & 95 \\ 92 & 94 & 96 & 98 & 99 & 99 & 98 & 97 \\ 94 & 96 & 99 & 101 & 103 & 103 & 102 & 101 \\ 95 & 97 & 101 & 104 & 106 & 106 & 105 & 105 \end{pmatrix}$$

Concentrating on the 1st row (say r1) our transformation process will occur in three steps. The first step is to group all of the columns in pairs: [88, 88], [89, 90], [92, 94], [96, 97] We replace the first 4 columns of r1 with the average of these pairs and replace the last 4 columns of r1 with half of the difference of these pairs.

We will denote this new row as r1h1= [88 89.5 93 96.5 0 −0.5 −1 −0.5]

The first 4 entries are called the approximation coefficients and the last 4 are called detail coefficients.

Next, we group the first 4 columns of this new row: [88, 89.5], [93, 96.5] and replace the first 2 columns of r1h1 with the average of the pairs and the next 2 columns of r1h1 with half of the difference of these pairs. We leave the last 4 rows of r1h1 unchanged. We will denote this second new row as r1h1h2= [88.75 94.75 −0.75 −1.75 0 −0.5 −1 −0.5]

Finally, our last step is to group the first 2 entries of r1h1h2 together: [88.75, 94.75] and replace the first column of r1h1h2 with the average of the pairs and the second column of r1h1h2 with half of the difference of these pairs. We leave the last 6 rows of r1h1h2 unchanged. We will denote this last new row as r1h1h2h3= [91.75 −3 −0.75 −1.75 0 −0.5 −1 −0.5]

We then repeat this process for the remaining rows of A. After this, we repeat this same process to columns of A, grouping rows in the same manner as columns. The resulting matrix is:

$$\begin{pmatrix} 96 & -2.0312 & -1.5312 & -0.2188 & -0.4375 & -0.75 & -0.3125 & 0.125 \\ -2.4375 & -0.0312 & 0.7812 & -0.7812 & 0.4375 & 0.25 & -0.3125 & -0.25 \\ -1.125 & -0.625 & 0 & -0.625 & 0 & 0 & -0.375 & -0.125 \\ -2.6875 & 0.75 & 0.5625 & -0.0625 & 0.125 & 0.25 & 0 & 0.125 \\ -0.6875 & -0.3125 & 0 & -0.125 & 0 & 0 & 0 & -0.25 \\ -0.1875 & -0.3125 & 0 & -0.375 & 0 & 0 & -0.25 & 0 \\ -0.875 & 0.375 & 0.25 & -0.25 & 0.25 & 0.25 & 0 & 0 \\ -1.25 & 0.375 & 0.375 & 0.125 & 0 & 0.25 & 0 & 0.25 \end{pmatrix}$$

This resulting matrix has several 0 entries and most of the remaining entries are close to 0. This is a result of the differencing and the fact that adjacent pixels in an image generally do not differ by much.

Proposed Algorithm for determining edge of a sample footprint using DWT (Haar function)

Step1. Read the image.

Step2. Convert the image from RGB to Grayscale format.

Step3. Perform median filtering of the image.

Step4. Compute a global threshold level to convert the image to binary.

Step5. Convert Grayscale image to binary.

Step6. Perform single-level 2D wavelet decomposition.

Step7. Create an array of zeros with the specified underlying class.

Step8. Perform single-level wavelet reconstruction.

Step9. Sub-plot the original and final images.

Step10. End.

Output

The output is shown in the following figures.

Fig 6: Sample Image.

Fig 7: Grayscale Image.

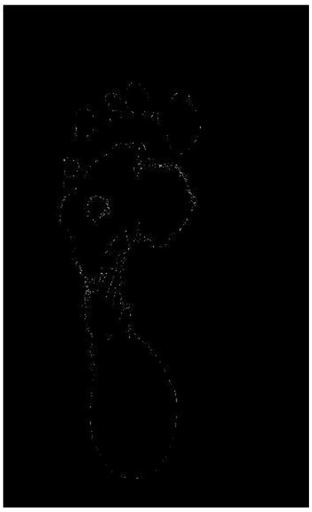

Fig 8: Resultant Image.

CONCLUSIONS

Image segmentation is the first step in image analysis. One of the methods to perform image Segmentation is Edge detection. In this paper, we propose a novel method image edge detection using 2-D Discrete Wavelet Transform for foot print.

REFERENCES

1. Rafael C. Gonzalez, and Richard E. Woods, Digital Image Processing, 2nd ed., Beijing: Publishing House of Electronics Industry, 2007.

2. John Canny , ‘A computational approach to edge detection’, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986; PAMI-8(6): 679-698.
3. G. Kaur and B. Singh,’Intensity Based Image Segmentation Using Wavelet Analysis and Clustering Techniques’, IEEE Seminar on AMTDC, 2002; 11/1-11/4.
4. N. R. Pal and S. K. Pal, ‘A Review on Image Segmentation Techniques’, Pattern Recognition, 1993; 26(9): 1277-1294.
5. Raman Maini and Dr. Himanshu Aggarwal, ‘Study and Comparison of Various Image Edge Detection Techniques’, IJIP, 2009; 3(1): 1-12.
6. Mamta Juneja, Parvinder Singh Sandhu, ‘Performance Evaluation of Edge Detection Techniques for Images in Spatial Domain’, IJCTE, 2009; 1(5): 614-621.
7. Gaurav K. Srivastava, Rohit Verma, Ruchika Mahrishi, and Siddavatam Rajesh,‘A Novel Wavelet Edge Detection Algorithm for Noisy Images’, International Conference on ICUMT, 2009; 1-8.
8. E. Brannock and M. Weeks, ‘A Synopsis of Recent Work in Edge Detection using the DWT’, IEEE Digital Object Identifier, 2008; 515-520.