COMPUTOLOGY: Journal of Applied Computer Science and Intelligent Technologies
Volume 1, Issue 2, Jul-Dec 2021, pp. 19-26
DOI: 10.17492/computology.v1i2.2103
https://www.journalpressindia.com/cjacsit

A Feature Extraction Procedure for Gray Scale Image

*Sumit Kumar**

ABSTRACT

In content based image retrieval (CBIR) system, its main concern is to search the relevant images from the large volume of image repository in reasonable time for their realization in real time application. The performance and effectiveness of CBIR scheme, is directly associated with construction of small dimensional as well as salient image features respectively. So, in this paper, we have carried out the image retrieval process with small dimensional salient image components or features as compare to the original image size and the image retrieval accuracy has been improved due to the consideration of local information rather than global information of image data. In works starts with the quantization scheme as all the pixels are not important of feature extraction. As, color information is missing in gray images hence we have deep extracted texture and shape features keeping in mind the directionalities and geometry of the image. To extract the texture, LBP and GLCM is employed and to capture the geometry at different level, we have deployed adaptive tetrolet transformation and EDH. This scheme has been validated through several benchmark databases.

Keywords: Content-Based Image Retrieval (CBIR); EDH; LBP; GLCM; Tetrolet;

1.0 Introduction

In this digital generation, most people have access to smartphones, digital camera, and many other devices, which generates a huge amount of multimedia data. Today where everyone is keen to click selfies around the globe, images have become an essential fraction of multimedia data. This generated data is stored either locally or on any third-party remote server. But, in both cases, one will might use it in near future. So, retrieving similar images to a query image becomes a big question. In the initial phase of image retrieval, it is carried out using a descriptive way, i.e., each image has been associated with a description according to the owner of the images or by the CBIR service provider. This practice has been obsoleted due to some reasons. First of all, describing the currently available volume of images is next to impossible. The time taken in this process could be unexpected. Secondly, each human has his/her thinking process. Like the picture given in Figure 2.1, one may be described as a mountain, snowy, Pictionary, and many more. This process is called text-based image retrieval (TBIR) [?]. So, to overcome this, researchers and academicians start focusing on retrieval systems based on the features of the image. This technique is formalised as content-based image retrieval (CBIR) [?]. Researchers have been using CBIR for more than two decades. In CBIR, primitive visual image features are extracted from all the images of the designated image database. During the initial phase, only one image features among color, texture, and shape [?] is considered.

**Research Scholar, Indian Institute of Technology, Dhanbad, Jharkhand, India (sumitvarshney68@gmail.com)*

But, later, people started to incorporate various features in different combinations and combined them as a feature vector for an image. The same set of features is extracted from all the images used in the retrieval scheme. In CBIR, image features are extracted in two ways, global and local. The selected visual feature is extracted from the entire image for global feature extraction. Though this type of extraction is fast, the resulted features do not closely associate with the image's actual essence. So, local image features became necessary. In local image feature extraction, initially, the image is divided into numbers of non-overlapping blocks, and then from each block, the intended feature is independently extracted, and finally, all the values are combined to form the feature vector. Since local image feature extraction takes a bit of time, there is always a trade-o_ between the time required and the system's efficiency. Therefore, the selection of the features becomes very tricky. The CBIR process can be divided into two phases. The first is the feature extraction, and the second is the similarity measurement. The user selects an appropriate similarity measure to retrieve similar images based on a query image in similarity measurement.

1.1 Motivation

But, we have encountered some of the images that are gray. They do not posses any color information. Therefore, we have to design a CBIR system which revolves around only texture and shape features but at the same time provide comparable results. In this work, to extract the texture features, we have deployed gray level co-occurrence matrix (GLCM) [] and LBP []. The reason of selecting these two approaches is that GLCM will capture information according to various on several directions and LBP on various orientations. For shape feature extraction, we have employed initially adaptive tetrolet transformation which have the capability to capture local structure and later, we have used edge descriptive histogram for overall object detection based on Canny edge detector. Later, all extracted features are combined to form the image feature vector. The same process has been carried out on each image of the respective image database to form the feature database.

1.2 Contribution of the Paper

The major contributions of this paper is as follows:

1. Texture and shape features are incorporated.
2. Both directionality and orientation based features are employed for texture information.
3. Local geometry as well as entire object detection has taken place.
4. The proposed scheme has been validated on several benchmark image databases.

2.0 Preliminaries

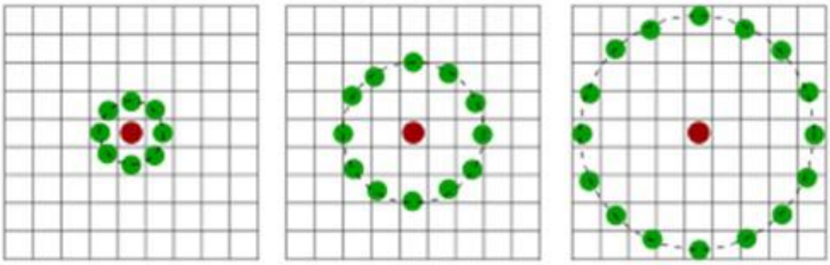
In this section, we have discussed briefly the techniques that are indulged in various feature extraction processes.

2.1 Local Binary Pattern

LBP [?] is a kind of visual descriptor employed for grouping in PC vision. LBP is the specific case of the Surface Range illustration proposed in 1990. Since 1990, LBP has been observed to be an intense module for texture ordering. It has additionally been resolved that when LBP is joined to Histogram of oriented gradients (HOG) [?] descriptor, it add to the recognition performance remarkably on few datasets. A correlation of a few deviations of the first LBP in the arena of foundation subtraction was brought on 2015 by Silva et al. A complete study of the various executions of LBP can be found in Bouwmans et al. The LBP feature vector, in its most straightforward shape, is done in different way:

- Separate the analysed window into cells (e.g. 16x16 pixels for every cell)
- For every pixel in a cell, contrast the pixel with every one of its 8 neighbors (to its left side best, left-center, left-base, right-top, and so forth.). Take after the pixels along a circle, i.e. clockwise or counter-clockwise.
- Where the center pixel’s value is taken if the center value is greater than the neighbor’s value, express ”0”. Else, state ”1”. This results in 8-digit binary number (which is typically changed over to decimal gradually).
- Histogram is computed, over each cell value, of the recurrence of each ”number” happening (i.e., every blend of which pixels are smaller and greater than the center). This histogram can be viewed as a 256-dimensional feature vector.
- Alternatively standardize the histogram.
- Histograms of all cells which results in the form of feature vector for the whole window.

Figure 1: Three Neighborhood Cases Used to Characterize a Texture and LBP is Calculated



2.2. Mid-rise quantization

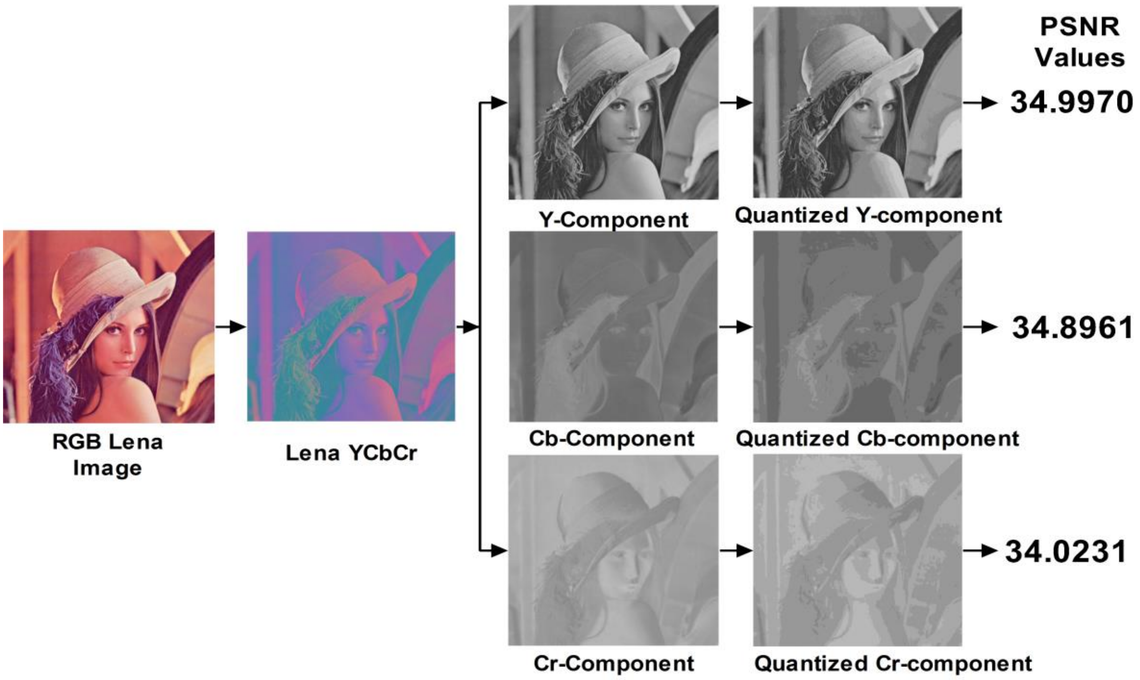
In this proposed work, we have used 16-bin quantization, so there are a total of 16 groups with intensities 0-15,16-31,32-48, and so on.

Figure 2: Various Quantized Value

$$Q = \begin{cases} 8 & \text{if } q \in \{0-15\} \\ 24 & \text{if } q \in \{16-31\} \\ 40 & \text{if } q \in \{32-47\} \\ 56 & \text{if } q \in \{48-63\} \\ 72 & \text{if } q \in \{64-79\} \\ 88 & \text{if } q \in \{80-95\} \\ 104 & \text{if } q \in \{96-111\} \\ 120 & \text{if } q \in \{112-127\} \\ 136 & \text{if } q \in \{128-143\} \\ 152 & \text{if } q \in \{144-159\} \\ 168 & \text{if } q \in \{160-175\} \\ 184 & \text{if } q \in \{176-191\} \\ 200 & \text{if } q \in \{192-207\} \\ 216 & \text{if } q \in \{208-223\} \\ 232 & \text{if } q \in \{224-239\} \\ 248 & \text{otherwise} \end{cases}$$

In figure 2, there are three images. The first one is the Y-component, the second one is the Cb-component, and the third one is the Cr-component of a Lena image and the second column showing their respective quantized image, and the third column is representing their PSNR value. PSNR value showed that reconstructed image is adequately similar to the reference image.

Figure 3: Quantized Image with their PSNR Values



2.3 Tetrolet transformation

The Discrete wavelet transformation (DWT) is applied along rows and columns only so that analysis can take place in only two directions. Still, only two directions cannot cover the complete geometry of an image. So, to produce optimal results several transform domain tools like curvelets [?], counterlets [?] and directionlets [?] to incorporate local geometry of an image. In this proposed methodology, we have adopted tetrolet transformation for efficient representation of an image. Tetrolet is a special type of traditional Haar wavelet [?].Tetrominoes are introduced by Golomb [?]. Jens Krommweh [?] proposed tetrolet transformation, which incorporates tetrominoes to represent an image efficiently.

The low-pass sub-band of an image I^{r-1} is divided into blocks Bi, j of size $4 \times 4,\ i, j = 0, 1, \ldots \frac{N}{4} - 1$
For each block Bi, j all 117 possible covering c=1,2,3....117 are implemented and for each tiling c^* 4 low-pass sub-bands and 12 high-pass sub-bands are obtained. Low-pass sub-bands at each decomposed level is extracted as follows:

$$A^{r,(c)} = (I^{r,(c)}[S])_{s=0}^{3}$$

$$I^{r,(c)}[S] = \sum_{(m,n)\in I_s^{(c)}} \in [0, L(m,n)] I^{r-1}[m,n]$$

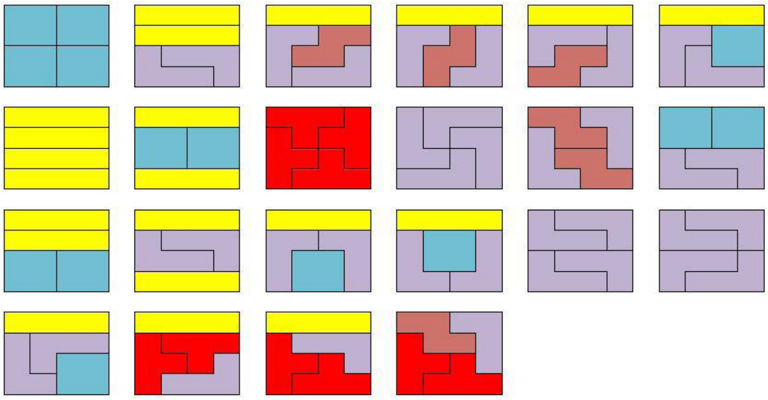
For the further decomposition, only low-pass sub-bands is considered. At the same time, all high pass sub bands are individually stored at each level for texture analysis.

2.4 Edge histogram descriptor

Edge Histogram Descriptor(EHD) provides information regarding the edge distribution using the histogram of the local edge's direction [?]. EHD shows 5 di_erent types of edges of an image by dividing it into non-overlapping

Figure 4: 5 Basic Tetrominoes, O-I-T-S-L tetrominoes

Figure 5: 22 Fundamental Tiling for 4 _ 4 board



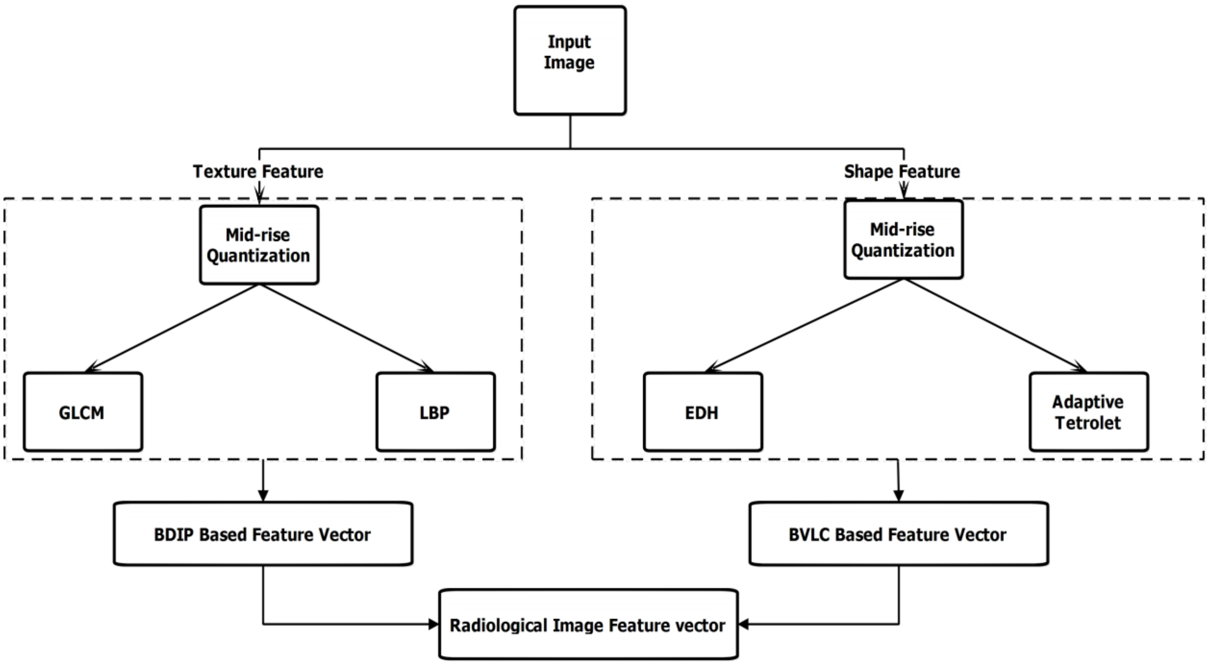
blocks. Each block is named as a sub-image. As described by Figure: 6, each image is divided into 4 _ 4 blocks so a total of 16 sub-images. Edges of each sub-image is divide into 5 categories, (i) vertical (ii) horizontal (iii) 450 (iv) 1350 (v) non-directional. So, there are a total of 16 sub-images and each sub-image has 5-corresponding edges so a total 80 bins histogram is generated.

Figure 6: Edge Histogram Descriptor

3.0 Proposed System

CBIR is one of the fatest growing area of research where various features like color, texture, and shape are extracted from the image in different ways. As, we have discussed earlier, in this paper we are considering only the gray scale images so here we have only texture and shape features. It is quite obvious that local image features are more associated with the actual content but its computational time is more hence to keep all these in mind, authors have proposed a novel work where global and local features are combinedly used to provide comparable results. The overall feature extraction process has been divided into two parts. In the first part, texture feature are extracted based on LBP and GLCM/. Second part comprises of shape features based on Tetrolet transformation and EDH. Lastly all the extracted features are put together to form the final feature vector. This process has been briefly illustrated in the Figure 7. Various Algorithmic Steps involved in the process are described in Algo 1.

Figure 7: Feature Extraction Process



Algorithm 1 Feature Extraction.
Input: Input Gray Image Q_1.
Output: Final feature vector f_{v1}.
Parameter: Size of QI is M * N,
Select a query input image QI of size M * N.

Shape Feature Extraction:
Initialize texture feature vector $f_{vs} = 0$.
Apply n round of adaptive tetrolet trasformation.
Compute various statistical parameters to get $f_{vs} = [ADT_f]$.
Apply EDH on the entire image to get $f_{vs} = [EDH_f, ADT_f]$.

Texture Feature Extraction:
Initialize shape feature vector $f_{vT} = 0$.
Apply LBP with the window size of 3 * 3.
Compute various statistical parameters to get $f_{vT} = [LBP_f]$.
Apply GLCM with 4 parameters on the entire image to get fvT = [$GLCM_f$, LBP_f].
Concatenate the texture and shape feature vectors to compute final feature vector as: f vI = [fvT, fvS].

4.0 Results and Discussion

This process has been carried out in MATLAB environment using a system with Intel(R) Core(TM) i7-4770 CPU @ 3.40 GHz, 4 GB RAM. In this experiment, we have drawn results based on Brodatz available at [http://sipi.usc.edu/database/database.php?volume=rotate], STex [?] image databases. The brief details of these databases are as follows:

(i) Brodatz database has a total of 1456 images from 13 categories and the size of each image is 128 _ 128. (ii) Salzburg texture image database (STex) has 476 images of size 512_512. Then each image is divided into 128_128 hence from one image there will be total 16 images is formed which produces the cardinality of database to 7616. Here, the categories like Buildings, Grass, Hair, Leaf, Leather etc. have images varies from 32 to 1232.

The authenticity of the proposed system has been tested using three well-known and accepted parameters which are precision, recall, and F-score. Precision measures the number of correctly retrieved image out of all retrieved image. Recall can be described as it let you know how many correct images are retrieved from all the images of a particular category in the respective database. They can be expressed as:

$$precision = \frac{X \cap R}{X}$$

$$recall = \frac{X \cap R}{R}$$

F-score is the harmonic mean of precision and recall. It gives you a value which reflect the overall performance of the system. This means that higher the value higher is the retrieval performance. The mean average precision for the Brodatz and STex image databases are 96:38 and 57:26 respectively. In the Table 1, we have compared our results with other similar schemes. In the Table 2, we have illustrated the results comparision based on mAP of STex image database.

$$F\_Score = \frac{2 \times precision \times recall}{(precision + recall)}$$

Table 1: Comparison of Brodatz Dataset on the Basis of mAP

S.N.	Methods	Average Precision
1.	Raghuwanshi ([?])	77.41
2.	Backes ([?])	95.50
3.	Guo ([?])	82.25
4.	Backes ([?])	95.75
5.	Kaya ([?])	94.41
6.	Proposed	**96.38**

Table 2: Comparison of STex Database on the Basis of mAP

S.N.	Method	MAP
1.	LBP	42.69
2.	CND	50.87
3.	DLEP	47.55
4.	LTrP	47.00
5.	LBP+LNDP	53.78
6.	Proposed	57.26

References

1. Filippini, Massimo, and Lester C. Hunt. (2011) “Energy demand and energy e_ciency in the OECD countries: a stochastic demand frontier approach.” Energy Journal 32 (2): 59–80.
2. Filippini, Massimo, and Lester C. Hunt. (2012) “US residential energy demand and energy e_ciency: A stochastic demand frontier approach.” Energy Economics 34 (5): 1484–1491.
3. Weyman-Jones, Thomas, J´ulia Mendonc¸a Boucinha, and Catarina Feteira In´acio. (2015) “Measuring electric energy e_ciency in Portuguese households: a tool for energy policy.” Management of Environmental Quality: An International Journal 26 (3): 407–422.
4. Saunders, Harry (2009) “Theoretical Foundations of the Rebound E_ect”, in Joanne Evans and Lester Hunt (eds) International Handbook on the Economics of Energy, Cheltenham, Edward Elgar
5. Sorrell, Steve (2009) “The Rebound E_ect: definition and estimation”, in Joanne Evans and Lester Hunt (eds) International Handbook on the Economics of Energy, Cheltenham, Edward Elgar