CONTENT BASED IMAGE RETRIEVAL USING LOCAL TETRA PATTERN

S. ANITHA¹, A. JEEVA², NIVEDITHA. R. DAS³, K. YOHESWARI⁴ & P. DEVI⁵

¹,²,³,⁴ Sri Ramakrishna Institute of Technology, Anna University, Chennai, India
³ Assistant Professor, Sri Ramakrishna Institute of Technology, Anna University, Chennai, India

ABSTRACT

Local tetra pattern (LTrP) is used for creating a new retrieval algorithm for managing the large database. It is necessary to develop appropriate information systems to efficiently manage these collections. The most relevant method to manage the large database is Content Based Image Retrieval (CBIR) system. The standard local binary pattern (LBP) and local ternary pattern (LTP) encode the relationship between the referenced pixel and its surrounding neighbors by computing gray-level difference. LTrP encodes the relationship between the referenced pixel and its neighbor pixel based on the directions that are calculated using the first-order derivatives in vertical and horizontal directions. It presents a statistical view of the texture retrieval problem by combining the two related tasks, namely feature extraction (FE) and similarity measurement (SM). And also compute the nth order local tetra pattern using (n-1)th order in horizontal and vertical derivatives for efficient CBIR. The performance of the proposed method is compared with the LBP, the local derivative patterns, and the LTP based on the results obtained using benchmark image database. Performance analysis shows that the proposed method improves the retrieval result from 70.34% to 75.9% in terms of average precision and from 44.9% to 48.7% in terms of average recall and from 79.97% to 85.30% in terms of average retrieval rate respectively, as compared with the standard LBP.

KEYWORDS: Content-Based Image Retrieval (CBIR), Local Binary Pattern (LBP), Local Tetra Patterns (LTrPs), Feature Extraction (FE), Similarity Measurement (SM)

INTRODUCTION

The rise of interest in techniques for retrieving images on the basis of automatically derived features such as color, texture and shape, a technology now generally referred to as Content-Based Image Retrieval (CBIR). After a decade of intensive research, CBIR technology is now beginning to move out of the laboratory and into the marketplace, in the form of commercial products like QBIC and Virage. Interest in the potential of digital images has increased enormously over the last few years, fuelled at least in part by the rapid growth of imaging on the World-Wide Web Users in many professional fields are exploiting the opportunities offered by the ability to access and manipulate remotely stored images in all kinds of new and exciting ways.

CBIR operates on a totally different principle from keyword indexing. Primitive features characterizing image content, such as color, texture and shape, are computed for both stored and query images, and used to identify the 20 stored images most closely matching the query. Semantic features such as the type of object present in the image are harder to extract, though this remains an active research topic. Video retrieval is a topic of increasing importance here, CBIR techniques are also used to break up long videos into individual shots, extract still key frames summarizing the content of each shot, and search for video clips containing specified types of movement.

The LBP operator on facial expression analysis and recognition is successfully reported in. Furthermore, the LBP is incorporated into multiscale heat-kernel face representation for the purpose of capturing texture information of the face appearance. Face image is decomposed into different scale and orientation responses by convolving with multiscale and
multiorientation Gabor filters. In the second phase, BP analysis is used to describe the neighboring relationship not only in image space but also in different scale and orientation responses.

Due to the discriminative power and computational simplicity, LBP texture operator has become a popular approach in various applications. It can be seen as a unifying approach to the traditionally divergent statistical and structural models of texture analysis. Local Derivative pattern explains the feasibility and effectiveness of using high-order local patterns for face representation. An LDP operator is proposed, in which based on a binary coding function (n-1)th order derivative is calculated. The LBP, the LDP, and the LTP extract the information based on the distribution of edges, which are coded using only two directions (positive direction or negative direction). Thus, it is evident that the performance of these methods can be improved by differentiating the edges in more than two directions. This observation has motivated us to propose the four direction code, referred to as local tetra patterns (LTrPs) for CBIR.

In a second-order LTrP that is calculated based on the direction of pixels using horizontal and vertical derivatives. Our method is different from the existing LDP in a manner that it makes use of 0 and 90 derivatives of LDPs for further calculating the directionality of each pixel. Finally, the generalized nth-order LTrP operator has been presented by using (n-1)th-order derivatives. The performance of our method is compared with the LBP, the LDP, and the LTP by conducting three experiments on different image database. Similar to LDP, in order to compare our method with the LBP, we consider the LBP as a non-directional first order local pattern called the first-order LTrP.

LOCAL PATTERNS

Local Binary Pattern

Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. Another important property is its computational simplicity, which makes it possible to analyze images in challenging real-time settings.

Local Derivative Pattern

Local Derivative pattern explains the feasibility and effectiveness of using high-order local patterns for face representation. An LDP operator is proposed, in which based on a binary coding function (n-1)th order derivative is calculated. LBP encodes all-direction first-order derivative binary result while LDP encodes the higher-order derivative information which contains more detailed discriminative features that the first-order local pattern (LBP) cannot obtain from an image.

Local Ternary Pattern

LBP’s are resistant to lighting effects in the sense that they are invariant to monotonic gray-level transformations, and they have been shown to have high discriminative power for texture classification. When using LTP for visual matching, we could use valued codes, but the uniform pattern argument also applies in the ternary case. For simplicity, the experiments below use a coding scheme that splits each ternary pattern into its positive and negative halves, subsequently treating these as two separate channels of LBP descriptors for which separate histograms and similarity metrics are computed, combining the results only at the end of the computation.

Local Tetra Pattern

It is evident that the performance of these methods can be improved by differentiating the edges in more than two directions. This observation has motivated us to propose the four direction code, referred to as local tetra patterns (LTrPs) for CBIR. The local tetra patterns (LTrPs) describes the spatial structure of the local texture using the direction of the
centre gray pixel It encodes the relationship between the referenced pixel and its neighbors, based on the directions that are calculated using the first-order derivatives in vertical and horizontal directions.

PROPOSED SYSTEM

Images from the Corel database have been used. This database consists of a large number of images of various contents ranging from animals to outdoor sports to natural images. These images have been pre-classified into different categories each of size 100 by domain professionals. For our experiment, we have collected 80 images to form database DB1. These images are collected from ten different domains, namely, Africans, beaches, buildings, buses, dinosaurs, elephants, flowers, horses, mountains, and food. In this experiment, each image in the database is used as the query image.

For these images, performed the image preprocessing. It includes the resizing all the images in to a standard image size 256X256 by using the function ‘imresize’. Then these images in RGB are converted to gray scale image by using the function ‘rgb2gray’. After these processes the noise in these images are removed and for smoothing the image we can even use edge detection and thinning process. For edge detection we use canny edge detection since it is the most efficient edge detection process.

Now the central pixel position of image is found out by using the equation

$$\frac{(N+1)}{2}$$

where N represents the number of pixels present in the image. Hence by this process the central pixel and its direction are calculated and also get the neighbor pixels in all four directions. The 8bit tetra pattern for each central pixel is found. Then separate all patterns into four parts based on the direction of centre pixel. Finally, the tetra patterns for each part (Direction) are converted to three binary patterns. Let the direction of Centre pixel obtained using equation 8 be 1. In order to get the direction of central pixel value, we need to apply the first order derivative in both vertical and horizontal direction i.e., 0 and 90 degree. The first-order derivatives along 0 and 90 directions are denoted as (1)

$$I^h_v(g_p), \theta = 0; 90^\circ$$

-------- (1)

Let gc denote the centre pixel in I, and let gh and gv denote the horizontal and vertical neighbourhoods of gc, respectively. Then, the first-order derivatives at the centre pixel can be written as the equation (2)

$$I^h_h (g_c) = I(g_h) - I(g_c)$$

$$I^h_v (g_c) = I(g_v) - I(g_c)$$

-------- (2)
By finding the position of central pixel value, we can find out the direction of this central pixel value by applying the first order derivative to this pixel. And the direction of central pixel can be calculated by the equation (3)

\[
I_{B_1}^{L}(g_c) = \begin{cases} 
1, & I_{B_1}^{L}(g_c) \geq 0 \text{ and } I_{B_2}^{L}(g_c) \geq 0 \\
2, & I_{B_1}^{L}(g_c) < 0 \text{ and } I_{B_2}^{L}(g_c) \geq 0 \\
3, & I_{B_1}^{L}(g_c) < 0 \text{ and } I_{B_2}^{L}(g_c) < 0 \\
4, & I_{B_1}^{L}(g_c) \geq 0 \text{ and } I_{B_2}^{L}(g_c) < 0
\end{cases}
\]

--- (3)

Fig 2 illustrates the possible local pattern transitions resulting in an LTrP for direction “1” of the center pixel. The LTrP is coded to “0” when it is equal to the direction of center pixel, otherwise coded in the direction of neighborhood pixel. Using the same analogy, LTrPs are calculated for center pixels having directions 2, 3, and 4. After identifying the local pattern PTN (the LBP, the LTP, the LDP, or the 13-binary-pattern form LTrP), the whole image is represented by building a histogram using

\[
H_{\mu}(I) = \frac{1}{N_x \times N_y} \sum_{k=1}^{N_x} \sum_{\mu=1}^{N_y} \#f_{\mu}(PTN(J, k), I)1_{\mu \in [0, P(\mu - 1) + 2]} 
\]

--- (4)

\[
f_{\mu}(x, y) = \begin{cases} 
1, & I_{B_2}^{L}(g_c) - y \\
0, & x \text{ or } y
\end{cases}
\]

--- (5)

Where \(N_x \times N_y\) represent the size of the input image. An example of the second-order LTrP computation resulting in direction “1” for a center pixel marked with red has been illustrated in Fig. 1. When we apply first-order derivative in horizontal and vertical directions to the neighborhood pixel “8,” we obtain direction “3” and magnitude “9.2.” Since the direction of the center pixel and the direction obtained from the neighborhood pixel are not same, we assign value “3” to the corresponding bit of the LTrP. It can be seen that the magnitude of the center pixel is “6,” which is less than the magnitude of neighborhood pixel. Hence, we assign value “1” to the corresponding bit of the magnitude pattern. Similarly, the remaining bits of the LTrP and the magnitude pattern for the other seven neighbors are computed resulting in the tetra pattern “3 0 3 4 0 3 2 0” and the magnitude binary. Pattern “1 1 0 0 0 1 0 1.” Referring to the generated LTrP, the first pattern is obtained by keeping “1” where the tetra pattern value is “2” and “0” for other values, i.e., “0 0 0 0 0 0 1 0.” Similarly, the other two binary patterns “1 0 1 0 0 1 0 0” and “0 0 0 1 0 0 0 0” are computed for tetra pattern values “3” and “4,” respectively.

In the same way, tetra patterns for center pixels having directions 2, 3, and 4 are computed. Thus, with four tetra patterns, 12 binary patterns are obtained. The 13th binary pattern is obtained from the magnitude of the first-order derivative. Hence we get a histogram value for this magnitude pattern too. Then combine both the histogram values and we get a value. Now we need to extract feature vector from this histogram values to get the feature vector.

\[
D(q, DB) = \sum_{\mu=1}^{P} \left[ \frac{f_{\mu}(B_{q}) - f_{\mu}(B)}{f_{\mu}(B_{q}) + f_{\mu}(B)} \right]
\]

--- (6)

Now perform these steps for the query image. Compare the results of query image with the database images and then retrieve the images. The performance of the proposed method can be calculated in terms of average precision, average recall, and average retrieval rate (ARR). For query image \(I_q\), the precision is defined as
\[
P(I_t, n) = \frac{1}{|DB|} \sum_{i=1}^{|DB|} \mathbb{I} \left( \text{Rank}(I_t, I_q) \leq n \right)
\]

--- (7)

Where “n” indicates the number of retrieved images and |DB| is the size of the image database. \(\mathbb{I}(x)\) is the category of “x”. \(\text{Rank}(I_i, I_q)\) returns the rank of image \(I_i\) (for the query image \(I_q\)) among all images of |DB|.

Recall is defined as
\[
R(I_q, n) = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbb{I} \left( \text{Rank}(I_t, I_q) \leq n \right)
\]

--- (8)

Finally, the total average precision and the ARR for the whole reference image database are computed,
\[
P_{\text{ave}}(n) = \frac{1}{|DB|} \sum_{i=1}^{|DB|} P(I_t, n)
\]

----------- (9)

\[
\text{ARR} = \frac{1}{|DB|} \sum_{i=1}^{|DB|} R(I_t, n)
\]

----------- (10)

The average recall is also defined in the same manner.

**Advantages of the LTrP over Other Patterns**

The advantages of the LTrP over the LBP, the LDP, and the LTP can be justified with the help of three points.

- The LBP, the LDP, and the LTP are able to encode images with only two (either “0” or “1”), two (either “0” or “1”), and three (“0”, “1”, or “1”) distinct values, respectively. However, the LTrP is able to encode images with four distinct values as it is able to extract more detailed information.

- The LBP and the LTP encode the relationship between the gray value of the center pixel and its neighbors, whereas the LTrP encodes the relationship between the center pixel and its neighbors based on directions that are calculated with the help of (n-1) th-order derivatives.

- The LDP encodes the relationship between the (n-1) th-order derivatives of the center pixel and its neighbors in 0, 45, 90, and 135 directions separately, whereas the LTrP encodes the relationship based on the relationship of the center pixel and its neighbors, which are calculated by combining (n-1) th-order derivatives of the 0 and 90 direction.

**Figure 2: Calculation of Tetra Pattern Bits for the Center-Pixel Direction “1” Using The Direction of Neighbors**

Direction of (Red) the Center Pixel and (Cyan) that its Neighborhood Pixels
Figure 3: Example to Obtain the Tetra and Magnitude Patterns. For Generating A Tetra Pattern, The Bit is Coded with the Direction of Neighbor when the Direction of the Center Pixel and Its Neighbor are Different, Otherwise "0." For the Magnitude Pattern, the Bit is Coded with “1” when the Magnitude of the Center Pixel is Less than the Magnitude of its Neighbor, Otherwise “0”

SIMULATED RESULTS

The database images consist of several images. In that we are taking 80 images which are being classified into ten different groups with 8 images in each domain, namely Africans, beaches, buildings, buses, dinosaurs, elephants, flowers, horses, mountains, and food. Here each image of the dimension 256X384 is resized to the dimension 256X256. The given image central pixel position is 33025, neighborhood pixel is 69,62,150,3,3,153,3,3,tetra pattern value is ‘21000000’, histogram value for local tetra pattern is 1. The magnitude and the magnitude pattern by using local binary pattern is ‘11101011’ and also the histogram value for the magnitude is 1. The feature extraction and similarity measurement have been processed, finally the image was retrieved.
CONCLUSIONS

The LTrP encodes the images based on the direction of pixels that are calculated by horizontal and vertical derivatives. The standard local binary pattern (LBP) and local ternary pattern (LTP) encodes the relationship between the
referenced pixel and its surrounding neighbours by computing gray level difference. The performance improvement of the proposed method has been compared and has been detailed below.

- The average precision has significantly improved from 70.34%, 72.9%, and 73.4% to 75.9% on database.
- The average recall has improved from 44.9%, 45.8%, and 46.9% to 48.7%, on database.

Horizontal and vertical pixels have been used for derivative calculation. Results can be further improved by considering the diagonal pixels for derivative calculations in addition to horizontal and vertical directions. Due to the effectiveness of the proposed method, it can be also suitable for other pattern recognition applications such as face recognition, fingerprint recognition, etc.

REFERENCES


AUTHOR'S DETAILS

S. Anitha, pursuing the B.E degree Electronics And Communication Engineering from Sri Ramakrishna Institute Of Technology, Anna University Coimbatore. She is an active member of IETE. She has presented National & International Conference in various fields. She had done the project in the area of Digital Image Processing. Her field of interest is Digital Image Processing, Digital Electronics, VLSI Design.
A. Jeeva, pursuing the B.E degree Electronics and Communication Engineering from Sri Ramakrishna Institute Of Technology, Anna University Coimbatore. She is an active member of IETE. She has presented National & International Conference in various fields. She had done the project in the area of Digital Image Processing. Her field of interest is Digital Image Processing, Digital Electronics, Antenna and wave propagation.

Niveditha R. Das, pursuing the B.E degree Electronics and Communication Engineering from Sri Ramakrishna Institute Of Technology, Anna University Coimbatore. She has presented National & International Conference in various fields. She had done the project in the area of Digital Image Processing. Her field of interest is Digital Image Processing, Microprocessor.

K. Yoheswari, pursuing the B.E degree Electronics and Communication Engineering from Sri Ramakrishna Institute Of Technology, Anna University Coimbatore. She has presented National & International Conference in various fields. She had done the project in the area of Digital Image Processing. Her field of interest is Digital Image Processing, Digital Electronics.

P. Devi received the BE degree in Electronics and Communication Engineering from VLB Janakiammal College of Engineering and Technology, Anna University Chennai in 2009 and the ME degree in Communication Systems from Sri Krishna College of Engineering and Technology, Anna University Coimbatore in 2011. Currently working in Sri Ramakrishna Institute of Technology as an Assistant Professor in ECE Department. She has two year teaching experience. She has presented the papers in the national conference. Under her guidance, the final year students are doing their project in various fields. Her field of interest is Digital Image Processing.