Similar imageretrieval based on texture feature vector using Local Octal and Local Hexadecimal Pattern and comparison with Local Binary Pattern

Nitin Arora¹, Alaknanda Ashok², Shamik Tiwari³

¹Department of Computer Science & Engineering, Uttarakhand Technical University, Dehradun, India
²University of Petroleum & Energy Studies, Dehradun, India
³Department of Electrical Engineering, G. B. Pant University of Agriculture and Technology, Pant Nagar, India

³Department of Cloud Computing & Virtualization, School of Computer Science, University of Petroleum & Energy Studies, Dehradun, India

¹nitinarora47@gmail.com, ²alakn@rediffmail.com, ³shamik.tiwari@ddn.upes.ac.in

Corresponding Author: Nitin Arora

https://doi.org/10.26782/jmcms.2019.08.00046

Abstract

Local binary patterns (LBP) is a very powerful texture feature of an image. Many variants of LBP models are available and almost all of the derived models are based on the idea to calculate the difference of each central pixel in the 3 × 3 neighborhood matrix. Based on this difference is positive or negative, we replace neighborhood pixel intensity with 1 or 0 respectively and then convert obtained 0 and 1 pattern into a decimal value. In this paper, we propose modification of this idea, instead of using local binary pattern, local octal and local hexadecimal pattern is used. Local octal pattern (LOP) and the local hexadecimal pattern (LHP) is further tested on two different datasets of 100 images each of sizes 150 x 150 and the obtained results are compared with the state-of-art local binary pattern. For similarity measure, Euclidian distance and Manhattan distance is used. Results show that local octal pattern is superior over local hexadecimal pattern and the local binary pattern is superior over both local octal pattern and local hexadecimal pattern.

Keywords : Feature extraction, local binary pattern, texture feature, content based image retrieval, pixel, pixel intensity
I. Introduction

Nowadays, it is very easy to access a digital device and to click thousands of images. Some of these clicked images are very useful and one wants to keep them safe for memory purpose. Later on, when someone wants access the same set of images, it becomes very difficult task to look through all the thousands of image and search for some particular image. In previous era, for searching the images from a set of thousand images text based technique were used. Text based image retrieval (TBIR) is a method of retrieving a particular image from image database. TBIR is very time consuming technique because in this user need to go through all the images for searching some particular image. For efficient retrieval of image from an image dataset another technique were proposed in year 1994 and named as Content based image retrieval (CBIR) [XIV]. In CBIR technique particular image (Query image) is searched based on the main contents of the image. Any image can have three types of contents namely texture content, color content and shape content [I]. Local Binary Pair (LBP) [IX] is one of the existing texture descriptor. In LBP, we consider 3x3 window of image pixels and calculate LBP value of each pixel based on the intensity value variance between central pixel of 3x3 matrix and all its eight neighbouring pixels. In this paper we mainly focus of LBP and proposed two more texture descriptor Local Octal pattern (LOP) and Local Hexadecimal Pattern (LHP).

Related Work

Many researchers worked on different texture features and produced many techniques for image retrieval. We covered some of the major texture techniques in the sections. Ojala et al. [X] suggested Local Binary Pattern (LBP) for image retrieval. LBP is based on texture feature. LBP highly attracted the researchers because of its easy to use and the computational proficiency. LBP is also good in case of light changes. There are many application of LBP like image searching, image browsing, face detection etc. In Local Binary Pattern (LBP), a matrix of size 3x3 of an image is used, and then the central pixel of the window is subtracted from all its eight neighbors one by one pixel. LBP uses, 0 or 1 based on if the difference between central pixel values and neighbour pixel value is negative or positive respectively, to generate a binary pattern. Decimal value to this 0 and 1 binary string represents the binary value of the corresponding image pixel. By taking LBP as a standard many improvements have been done on it by many researchers. J. Ren et al. [V] proposed Relaxed Local Ternary Pattern. X. H. Han et al. [XV] proposed Robust local ternary patterns for texture categorization. Quyuan Lin et al. [X] proposed local ternary pattern based on path integral (pi-LTP). Z. Wang et al. [XVII] proposed enhanced local ternary patterns method. Jian Li et al. [VI] proposed Extended Gradient Local Ternary Pattern for Vehicle Detection. S. Murala et al. [XI] proposed local tetra pattern. B. Zhang et al. [II] proposed Local derivative pattern (LDP). Sima Soltanpour et al. [XII] proposed Multiscale depth local derivative pattern (MD-LDP). He Yonggang et al. [III] proposed Pyramid-Based Multi-structure Local Binary Pattern. Shiv Ram Dubey et al. [XIII] proposed Local Bit-Plane Decoded Pattern. Xiaoyong Bian et al. [XVI] proposed extended multi-structure local binary pattern. Jing Yi Tou et al. [VI] proposed One-dimensional Grey-level Co-occurrence Matrices (GLCM).
This paper is organized in different main headings as: In section I, we have discussed our proposed texture descriptor. In section II, the existing LBP texture descriptor and the proposed local octal pattern and local hexadecimal pattern have been discussed. Section III, describes our proposed system framework and two different similarity measure technique. Section IV describes datasets used and obtained results. The last section V, we concluded our paper and mention some future scopes.

Table 1: Different types of Local patterns

| Local Descriptor Name                                      | Application Area          | Reference                  |
|-----------------------------------------------------------|---------------------------|----------------------------|
| Local Binary Pattern (LBP)                                | Texture Classification    | (Ojala et al., 2002)       |
| Relaxed Local Ternary Pattern (R-LTP)                     | Face Recognition          | (J. Ren et al., 2013)      |
| Robust Local Ternary Pattern (RLTP)                        | texture categorization    | (X. H. Han et al., 2013)   |
| Local Ternary Pattern based on path integral (pl-LTP)      | Steganalysis              | (Qian Lin et al., 2016)    |
| Enhanced Local Ternary Pattern                            | Automatic Target Recognition in Infrared Imagery | (Z. Wang et al., 2014) |
| Extended Gradient Local Ternary Pattern                    | Vehicle Detection         | (Jian et al., 2015)        |
| Local Tetra Pattern                                       | Face Recognition          | (S. Murala et al., 2012)   |
| Local Derivative Pattern                                  | Facial Image Recognition  | (Zhang et al., 2010)       |
| Multiscale depth local derivative pattern                 | sparse representation based 3D face recognition | (Sima Soltanpour et al., 2017) |
| Pyramid-Based Multi-structure Local Binary Pattern.        | Smoke detection in Videos | (He Yonggang et al., 2010) |
| Local Bit-Plane Decoded Pattern.                          | Biomedical Image Retrieval | (Shiv Ram Dubey et al., 2016) |
| Extended multi-structure local binary pattern.            | Image Scene Classification | (Xiaoyong Bian et al., 2016) |
| One-dimensional Grey-level Co-occurrence Matrices (GLCM)  | Texture classification    | (Jing Yi Tou et al., 2008) |

Main Contribution

Many local descriptor patterns have been proposed and developed for image retrieval based on content till now. However, maximum of the current local patterns share the intensity of a central pixel in a 3x3 matrix with one of its 8 neighboring pixels at a time to encrypt it in binary form as shown in figure 1. In this paper our main contribution is that we calculated local octal pattern and local hexadecimal pattern and compared the results with state-of-art local binary pattern. For local octal pattern we compared the intensity of the central pixel in a 3x3 matrix with one of its neighboring pixels at a time and then modulo with 8 is calculated and the value 0, 1, 2, 3, 4, 5, 6 or 7 is assigned based on the calculated value. Same procedure is used for local hexadecimal pattern. In this, we compared the intensity of the central pixel in a 3x3 matrix with one of its neighboring pixels at a time and then the modulo with 16 is calculated and the value 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E is assigned based on the calculated value. For the similarity measure we used Euclidean distance and Manhattan distance. Proposed systems are tested on two different data sets.

II Local Patterns

Any image can have three types of main components namely texture feature, color feature and shape feature. Texture feature is one of the main feature components of...
any image. Local patterns are types of texture features. All local patterns share the intensity of a central pixel in a $3 \times 3$ matrix with one of its 8 neighboring pixels. This paper is more focused on local binary pattern.

**Local Binary Pattern**

Local Binary Pattern (LBP) texture feature is most frequently used texture descriptor. In LBP, a matrix of size $3 \times 3$ of an image is used, and then the central pixel of the window is subtracted from all its eight neighbors’ one by one pixel. LBP uses, 0 or 1 based on if the variance between central and neighbour pixel value is negative or positive respectively, to generate a binary pattern. Decimal value to this 0 and 1 binary string represents the binary value of the corresponding image pixel.

Mathematically Local Binary Pattern can be calculated as

$$
LBP \left( N, R \right) = \sum_{i=1}^{N} 2^{i-1} \times D \left( I_i, I_c \right)
$$

(1)

$$
D \left( I_i, I_c \right) = \begin{cases} 
1, & \text{if } I_i \geq I_c \\
0, & \text{Otherwise}
\end{cases}
$$

(2)

Where \( N \) is the number of neighboring pixels and \( R \) is radius. \( I_c \) and \( I_i \) is representing the intensity of central pixel and \( i^{th} \) neighbour pixel respectively. Calculation is LBP value is explained in figure 1.

![Figure 1](image-url)

Fig.1. Diagrams (i) to (vi) are presenting the calculation of Local Binary Pattern (LBP) value for a particular pixel (i) a $3 \times 3$ window with universal representations of the centred and its eight neighboring pixels (ii) an example of a $3 \times 3$ window with centred pixel’s intensity 40 and eight neighboring pixel’s intensities as shown (iii) pixels with centred and binary values 0 and 1 are assigned as a result based on sign of difference values (iv) specific weights binary digits 0 and 1 (v) multiplication with weights (vi) Addition of all the values to get LBP value.

**Local Octal Pattern**

For local octal pattern we compared the intensity of the central pixel in a $3 \times 3$ matrix with one of its neighboring pixels at a time and then modulo with 8 is calculated and the value 0, 1, 2, 3, 4, 5, 6 or 7 is assigned based on the calculated value. Mathematically Local Octal Pattern can be calculated as:

Copyright reserved © J. Mech. Cont.& Math. Sci.
Nitin Arora et al

561
LOP \((N, R) = \sum_{i=1}^{N} 8^{i-1} \times M(I_i, I_c)\)  

\[ M(I_i, I_c) = \begin{cases} 
(I_i - I_c) \mod 8, & \text{if } I_i \geq I_c \\
0, & \text{Otherwise} 
\end{cases} \]

Where \(N\) is the number of neighboring pixels and \(R\) is radius. \(I_c\) and \(I_i\) is representing the intensity of central pixel and \(i^{th}\) neighbour pixel respectively. Calculation is LOP value is explained in figure 2.

**Local Hexadecimal Pattern**

In this, we compared the intensity of the central pixel in a 3x3 matrix with one of its neighboring pixels at a time and then the modulo with 16 is calculated and the value \(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E\) is assigned based on the calculated value.

Mathematically Local Hexadecimal Pattern can be calculated as:

\[ \text{LHP} \,(N, R) = \sum_{i=1}^{N} 16^{i-1} \times H(I_i, I_c) \]
\[ H(I_i, I_c) = \begin{cases} (I_i - I_c) \% 16, & I_i \geq I_c \\ 0, & \text{Otherwise} \end{cases} \]  

Where \( N \) is the number of neighboring pixels and \( R \) is radius. \( I_c \) and \( I_i \) is representing the intensity of central pixel and \( i^{th} \) neighbour pixel respectively. Calculation is LBP value is explained in figure 3.

![Diagram](image)

**III. Proposed System Framework**

The proposed system framework is shown in figure 4. The complete system is composed of three phases. In phase 1, we are calculating feature vectors and distance measure (eq. 7 and eq.8) using LBP. In phase 2, we are calculating feature vectors and distance measure (eq. 7 and eq.8) using LOP and in phase 3, we are calculating feature vectors and distance measure (eq. 7 and eq.8) using LHP.
System Framework Algorithm

Proposed framework algorithm works in three phases. In phase 1 (step 1 to step 3), we calculated feature vectors and distance measure (eq. 7 and eq.8) using LBP. In phase 2 (step 4 to step 6), we calculated feature vectors and distance measure (eq. 7 and eq.8) using LOP and in phase 3 (step 7 to step 9), we calculated feature vectors and distance measure (eq. 7 and eq.8) using LHP. At last all the computed distances are compared in step 10.

Input: Set of images and query image
Output: Image similar to query image

Phase 1:

1. Calculate feature vector of all the image in the dataset using LBP texture descriptor
2. Calculate feature vector of query image using LBP texture descriptor
3. Calculate Euclidian distance and Manhattan distance between feature vectors.
Phase 2:
4. Calculate feature vector of all the image in the dataset using LOP texture descriptor
5. Calculate feature vector of query image using LOP texture descriptor
6. Calculate Euclidian distance and Manhattan distance between feature vectors.

Phase 3:
7. Calculate feature vector of all the image in the dataset using LOP texture descriptor
8. Calculate feature vector of query image using LOP texture descriptor
9. Calculate Euclidian distance and Manhattan distance between feature vectors.
10. Compare the distance calculated in step 3, step 6 and step 9.

Similarity Measure

For similarity measure many distance methods are available namely d1 distance, Euclidian distance, Manhattan distance, Canberra distance, Chi-square distance. In this paper we used Euclidian distance and Manhattan distance.

Euclidian Distance

Euclidian distance is linear distance between to features vectors. In case of two similar feature vectors Euclidian distance is 0. Mathematically Euclidian distances between two feature vectors can be calculated with the help of eq. 7.

$$D = \left( \sum_{j=1}^{n} \left| F_{dj} - F_{qj} \right|^2 \right)^{1/2}$$  (7)

Where, D is the distance and $F_{dj}$ is feature vector of $j^{th}$ image in database and $F_{qj}$ is $j^{th}$ query image.

Manhattan Distance

Manhattan distance is another way to measure the distance between two feature vectors. Mathematically Manhattan distances between two feature vectors can be calculated with the help of eq. 7.

$$D = \left( \sum_{j=1}^{n} \left| F_{dj} - F_{qj} \right| \right)$$  (8)

Where, D is the distance and $F_{dj}$ is feature vector of $j^{th}$ image in database and $F_{qj}$ is $j^{th}$ query image.
IV. Data Sets and Experimental Results

In this paper we used two different data sets WANG Data set SIMPSON Data set. Features vectors of these images are calculated using LBP, LOP and LHP. Euclidian distance and Manhattan distance are used to find the similarity difference between the query image and image dataset. In this we consider 5 images as number of output images.

Dataset 1

Dataset 1 (WANG Dataset) contains ten categories of 100 images each of size 50 x 50 [VII] (figure 5).

![Images of dataset 1](image1.jpg)

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. 5: Image (a) to (j) presenting different categories of images in used dataset 1.

The different categories of images available in used dataset 1 is described in table 2.

Table 2: Different types of classes and categories in used dataset 1

| Sr. No. | Class # | Image Category |
|---------|---------|----------------|
| 1       | Class 1 | African        |
| 2       | Class 2 | Beach          |
| 3       | Class 3 | Monument       |
| 4       | Class 4 | Buses          |
| 5       | Class 5 | Dinosaurs      |
| 6       | Class 6 | Elephant       |
| 7       | Class 7 | Rose           |
| 8       | Class 8 | Horse          |
| 9       | Class 9 | Snowy hills    |
| 10      | Class 10| Food           |
The output of LBP, LOP and LHP by using Euclidian distance matrix is shown in figure 6, figure 7 and figure 8 respectively.

**Fig. 6:** Showing output images using LBP technique and Euclidean distance method for Data Set 1

Figure 6 shows that LBP technique retrieving the correct image using Euclidean distance method. Euclidean distance between query image and the first retrieved images is 0.

**Fig. 7:** Showing output images using LOP technique and Euclidean distance method for Data Set 1

**Fig. 8:** Showing output images using LHP technique and Euclidean distance method for Data Set 1
Figure 7 shows that LOP technique retrieving the correct image using Euclidean distance method. Euclidean distance between query image and the first retrieved images is 0.

![Query Image](image)

| Im 1: 7961135146.6894 | Im 2: 7961135703.27908 |
|------------------------|------------------------|
| Im 3: 7961135730.3695   | Im 4: 7961135898.2809   |
| Im 5: 7961135974.01124  | Im 6: 7961136003.8508   |
| Im 7: 7961136042.1530   | Im 8: 7961136061.2284   |

Fig. 8: Showing output images using LHP technique and Euclidean distance method for Data Set 1

Figure 8 shows that LHP technique not retrieving the correct image using Euclidean distance method. Euclidean distance between query image and the first retrieved images is not 0.

The output of LBP, LOP and LHP by using Manhattan distance is shown in figure 9, figure 10 and figure 11 respectively.
Fig. 9: Showing output images using LBP technique and Manhattan distance for Data Set 1

Figure 9 shows that LBP technique retrieving the correct image using Manhattan distance method. Manhattan distance between query image and the first retrieved images is 0.
Fig. 10: Showing output images using LOP technique and Manhattan distance for Data Set 1

Figure 10 shows that LOP technique retrieving the correct image using Manhattan distance method. Manhattan distance between query image and the first retrieved images is 0.
Figure 11 shows that LHP technique retrieving the correct image using Manhattan distance method. Manhattan distance between query image and the first retrieved images is not 0.

**Dataset 2**

Dataset 2 (SIMPSON Data set) contains ten categories of fifty images each of size 50 x 50 (figure 12).

The different categories of images available in used dataset 2 is described in table 3.
Table 3: Different types of classes and categories in used dataset 2

| Sr. No. | Class # | Image Category          |
|---------|---------|-------------------------|
| 1       | Class 1 | Abraham Grampa          |
| 2       | Class 2 | Apu                     |
| 3       | Class 3 | Bart                    |
| 4       | Class 4 | Charles                 |
| 5       | Class 5 | Chief                   |
| 6       | Class 6 | Comic book guy          |
| 7       | Class 7 | Edna Krabappel          |
| 8       | Class 8 | Homer                   |
| 9       | Class 9 | Kent Brockman           |
| 10      | Class 10| Krusty                  |

The output of LBP, LOP and LHP by using Euclidian distance is shown in figure 13, figure 14 and figure 15 respectively.

![Query Image](image1)

![Im 1: 0.00000](image2)

![Im 2: 2405.29251](image3)

![Im 3: 2798.78683](image4)

![Im 4: 5656.19775](image5)

![Im 5: 6551.71179](image6)

![Im 6: 6797.38855](image7)

![Im 7: 8281.42437](image8)

![Im 8: 8561.75792](image9)

Fig. 13: Showing output images using LBP technique and Euclidean distance method for Data Set 2
Figure 13 shows that LBP technique retrieving the correct image using Euclidean distance method. Euclidean distance between query image and the first retrieved images is 0.

![Figure 13: LBP technique retrieving the correct image using Euclidean distance method.](image)

Fig. 14: Showing output images using LOP technique and Euclidean distance method for Data Set 2

Figure 14 shows that LOP technique retrieving the correct image using Euclidean distance method. Euclidean distance between query image and the first retrieved images is 0.

![Figure 14: LOP technique retrieving the correct image using Euclidean distance method.](image)

Fig. 15: Showing output images using LHP technique and Euclidean distance method for Data Set 2

![Figure 15: LHP technique retrieving the correct image using Euclidean distance method.](image)
Figure 15 shows that LBP technique not retrieving the correct image using Euclidean distance method. Euclidean distance between query image and the first retrieved images is not 0.

The output of LBP, LOP and LHP by using Manhattan distance is shown in figure 16, figure 17 and figure 18 respectively.

Fig. 16: Showing output images using LBP technique and Manhattan distance method for Data Set 2

Figure 16 shows that LBP technique retrieving the correct image using Manhattan distance method. Euclidean distance between query image and the first retrieved images is 0.

Fig. 17: Showing output images using LOP technique and Manhattan distance method for Data Set 2
Figure 17 shows that LOP technique retrieving the correct image using Manhattan distance method. Euclidean distance between query image and the first retrieved images is 0.

Fig. 18: Showing output images using LHP technique and Manhattan distance method for Data Set 2

Figure 18 shows that LHP technique retrieving the correct image using Manhattan distance method. Euclidean distance between query image and the first retrieved images is not 0. Table 4 shows the summary of obtained results using all the three techniques on both the datasets using Euclidean distance measure and Manhattan distance measure.
Table 4: Summary of the results obtained using all the three techniques on both the data sets.

| Technique used                     | Data set used          | Distance Matrix | Measured distance | Retrieved image |
|------------------------------------|------------------------|------------------|-------------------|-----------------|
| Local Binary Pattern (LBP)         | WANG dataset (dataset 1) | Euclidean        | 0                 | Correct         |
|                                    |                        | Manhattan        | 0                 | Correct         |
| SIMPSON dataset (dataset 2)        | Euclidean              | 0                | Correct           |
|                                    | Manhattan              | 0                | Correct           |
| Local Octal Pattern (LOP) (proposed) | WANG dataset (dataset 1) | Euclidean        | 0                 | Correct         |
|                                    |                        | Manhattan        | 0                 | Correct         |
| SIMPSON dataset (dataset 2)        | Euclidean              | 0                | Correct           |
|                                    | Manhattan              | 0                | Correct           |
| Local Hexadecimal Pattern (LHP) (proposed) | WANG dataset (dataset 1) | Euclidean        | Non zero          | Not correct     |
|                                    |                        | Manhattan        | Non zero          | Correct         |
| SIMPSON dataset (dataset 2)        | Euclidean              | Non zero         | Not Correct       |
|                                    | Manhattan              | Non zero         | Correct           |

V. Conclusion and Future Scope

In this paper we proposed two texture feature descriptor LOP and LHP and tested on two different datasets. The obtained results are compared with state-of-art LBP. For similarity measure, Euclidian distance and Manhattan distance is used. Results shows that local octal pattern is superior over local hexadecimal pattern and local binary pattern is superior over both local octal pattern and local hexadecimal pattern. The time taken to calculate the LBP, LOP and LHP for small images is almost same but, for large size images the time taken for LBP is very small in comparison with LOP and LHP. In future we can proposed more efficient and effective texture descriptor for producing better results. For, similarity measure, more distance measures can also be used.
References

I. A. Alaknanda, A. Nitin: ‘Content based image retrieval using Histogram and LBP’, International Journal of Communication System and Network Technology, vol. 5, No. 1, 2016, pp. 50-65

II. B. Zhang, Y. Gao, S. Zhao, J. Liu, "Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor", IEEE Trans. Image Process., vol. 19, pp. 533-544, 2010.

III. He Yonggang, Nong Sang, Changxin Gao, “Pyramid-Based Multi-structure Local Binary Pattern for Texture Classification”, Pattern Analysis and Applications 16(4):133-144, November 2010.

IV. Jian Li, Hanyi Du, Yingru Liu, Kai Zhang, Hui Zhou, “Extended Gradient Local Ternary Pattern for Vehicle Detection” IEEE 17th International Conference on Computational Science and Engineering, pp. 1882-1885, January 2015.

V. J. Ren, X. Jiang, J. Yuan, "Relaxed local ternary pattern for face recognition", IEEE International Conference Image Processing (ICIP), pp. 2-6, September, 2013.

VI. Jing Yi Tou; Yong Haur Tay; Phooi Yee Lau; "One-dimensional Grey-level Co-occurrence Matrices for texture classification," Information Technology, 2008. ITSim 2008, vol.3, no., pp.1-6, 26-28 Aug. 2008.

VII. N. Arora, A. Ashok, S. Tiwari, “Modified Local Binary Pattern Scheme using Row, Column and Diagonally aligned Pixel’s Intensity Pattern” International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 8, no. 5, pp. 771-779, March 2019.

VIII. Ojala, T., Pietikainen, M., Harwood, D.: ‘A comparative study of texture measures with classification based on feature distributions’, Pattern Recognition, 1996, 29, pp. 51–59

IX. Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7), 971–987.

X. Qiuyan Lin, Jiaying Liu, Zongming Guo, “Local ternary pattern based on path integral for steganalysis”, 2016 IEEE International Conference on Image Processing (ICIP), pp. 2737-2741 August 2016.

XI. S. Murala, R.P. Maheshwari, R. Balasubramanian, "Local tetra patterns: a new feature descriptor for content-based image retrieval", Trans. Image Process., vol. 21, no. 5, pp. 2874-2886, 2012.
XII. Sima Soltanpour, Q. M. Jonathan Wu, “Multiscale depth local derivative pattern for sparse representation based 3D face recognition”, IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 560-565, December 2017.

XIII. S. R. Dubey, S. Singh, and R. Singh, “Local bit-plane decoded pattern: A novel feature descriptor for biomedical image retrieval,” IEEE J. Biomed. Health Informat, vol. 20, no. 4, pp. 1139–1147, Jul. 2016.

XIV. Ying Liu, Dengsheng Zhang, et al.: ‘A survey of Content Based Image Retrieval with high-level semantics, Pattern Recognition, 40(1):262–282, January 2007.

XV. X. H. Han, G. Xu, Y. W. Chen, "Robust local ternary patterns for texture categorization", 2013 6th International Conference on Biomedical Engineering and Informatics, pp. 846-850, 2013

XVI. X. Y. Bian, C. Chen, Q. Du, and Y. X. Sheng, “Extended multistructure local binary pattern for high-resolution image scene classification,” in Proc. IEEE 36th Int. Conf. Geosci. Remote Sens. Symp., 2016, pp. 5134–5137.

XVII. Z. Wang, R. Huang, W. Yang, C. Sun, "An enhanced local ternary patterns method for face recognition", Proceedings of the 33rd Chinese Control Conference, pp. 4636-4640, July 2014