Journal of AI and Data Mining
Vol 7, No 2, 2019, 287-297 DOI: 10.22044/JADM.2018.6639.1787

Compressed Image Hashing using Minimum Magnitude CSLBP

V. Patil* and T. Sarode

Department of Computer Engineering, Thadomal Shahani Engineering College, Mumbai University, Mumbai, India.

Received 26 January 2018; Revised 23 February 2018; Accepted 06 April 2018
**Corresponding varshasp2977@gmail.com (V. Patil).*

Abstract
Image hashing allows compression, enhancement or other signal processing operations on digital images that are usually acceptable manipulations. Cryptographic hash functions are very sensitive to even single bit changes in image. Image hashing is a sum of important quality features in quantized form. In this paper, we propose a novel image hashing algorithm for authentication, which is more robust against various kinds of attacks. In the proposed approach, a short hash code is obtained using a minimum magnitude Center Symmetric Local Binary Pattern (CSLBP). The desirable discrimination power of image hash is maintained by modified Local Binary Pattern (LBP) based edge weight factor generated from gradient image. The proposed hashing method extracts texture features using the CSLBP. The discrimination power of hashing is increased by weight factor during the CSLBP histogram construction. The generated histogram is compressed to 1/4 of the original histogram by a minimum magnitude of CSLBP. The proposed method, has a two-fold advantage; first, it has small length, and second, it has an acceptable discrimination power. The experimental results are demonstrated by the hamming distance and the TPR, FPR, and ROC curves. Therefore, the proposed method successfully does a fair classification of content preserving and content changing images.

Keywords: *Authentication, CSLBP, LBP, Hashing, Histogram, Tampering.*

1. Introduction

Security of multi-media contents is important and also a challenge due to the growing demand for multi-media applications and spreading the use of multi-media information. Moreover, the digital media is constantly experiencing tremendous developments and advancements. Digital media is more advantageous due to its support for easy copying, sharing, and modifications which is not possible in the case of analog media. The reason behind the exponential growth in popularity is the ease of handling digital media. There are numerous techniques to deal with authentication of the original content. Watermarking is a popular method for authentication but it has its own limitations. It does not seem to be the a sustainable solution against a variety of attacks, especially if the algorithm works in spatial domain. Watermark embedding is an explicit activity and it becomes more cumbersome in case of video data. It also results in altering the original contents of an image, to some extent, in the frequency and spatial domain.

Hash provides hashing index for an efficient and fast searching and retrieval of information. In the same way, image hash is a compact code that is stored in image header, which allows an easy and efficient discrimination between the content change and content preserving of an image. Image hash code is in a binary form, which results in the fastest comparison with hamming distance on the receiver side.

In this paper, our focus is mainly on the texture features that work at the neighbourhood level. Image hash allows content preserving operations and does not support content changing operations. Although, cryptographic hashing and image hashing are similar concepts, cryptographic hashing is a strict approach that can produce different hashes even if a change occurs in a single bit. The cryptographic hashing is more

suitable for text data. Such a strict approach cannot be enforced on multi-media data, where changes in appearance are very common. Cryptographic hashing generates a fixed size hash code for any size of text data, whereas in image hashing, the generated hash code depends on image size as well as algorithm implementation.

The generated image hash is not inserted in the image data, rather it is stored in the image header. This results in retaining the original content of the image. To identify deviation from an original image, the hash code of the original image is compared with the hash code of the modified image. Apart from the compact size, other desirable properties of the hash are discrimination power to distinguish between content preservation and content change, localization of counterfeit area, and uniqueness or low collision probability.

2. Review of literature

Earlier study shows that, more focus is given on research works related to various features, domains, and matrix factorization methods. Less attention is paid to research works related to compact hash code. The previous method's focus is only on the quality feature extraction.

Coefficients in a transformed domain can be critical features and robust enough for content preserving, while being sensitive to content change operations. Sun et al. [1] have used the Contourlet HMT model, which senses inter-scale, inter-direction, and inter-location contourlet coefficients, that are steady to content preserving modifications and can be compressed. Srivastava et al. [2] have applied DCT on Radon transformed image for AC component extraction. From AC coefficients, statistical features are calculated to form a feature vector. Kang et al. [3] have utilized compressive sensing/sampling (CS) to make the hash smaller and secure. DWT is used as the basis matrix, and only the non-sparse data is considered. A similar approach based on compressed sensing was followed by Mo et al. [4], where they used 9/7 wavelet basis to find the sparse data in the frequency domain. For a compact and compressed representation, only non-sparse data is stored. Lei et al. [5] have combined the global features by DCT and the local features by the least-squares line (LSL) fitting of Discrete Wavelet Transform (DWT) coefficients. Guo and Hatzinakos [6] have combined discrete wavelet transform (DWT) and Radon transform. Frequency localization and shift/rotation invariant property of DWT and Radon are utilized for perceptual hash generation. Yu et al. [7] have selected DCT coefficients of low or middle frequency sub-bands and modeled as generalized Gaussian distribution (GGD). After this, maximum likelihood (ML) estimation is used to determine the shape feature. We have reviewed various methods that contribute to the local and global methods. Karsh et al. [8] have used the projected gradient nonnegative matrix factorization (PGNMF) for a shorter hash using a ring partition circular image. For local manipulation, salient regions are extracted. Zhao et al. [9] have combined the local (saliency map) and global (zernike moments) features for creating an image hash. Advantages of zernike moments are that they are uncorrelated, rotation invariant, and detect accurate detailed shapes. Saliency map focuses on local features such as texture and position of significant regions of the image. Soman and John [10] have combined the local and global features as haralic features and zernike moment features, respectively. The Haralick texture features describe 14 statistics for the local features that are calculated from the gray level co-occurrence matrix. A similar approach was followed by Sebastian et al. [11], where the local texture features were extracted from the Haralick, MOD-LBP, and global features by Zernike moments.

Wang et al. [12] have mined visual sensitive features from the Watson's visual model. These Watson's DCT coefficient forms block based features. SIFTs that are invariant to translation, rotation, scaling, and robust to content preserving, are used to extract the key point-based features. Lai et al. [13] have proposed a deep architecture for multi-label image retrieval. This method identifies each label and generates a hash for that group using the features of that label. Lv and Wang [14] have detected stable robust feature points using the SIFT and Harris detectors, and for the local features, the shape context has been used.

Tang and his colleagues have proposed various hashing approaches in [15-19]. A normalized image is generated from a pre-processed input image for a stable feature extraction. The proposed approach in [15] uses a ring partition with NMF. The approach proposed [16] uses the ring partition with an invariant vector distance, and the proposed approach [17] describes entropies of the ring partition. The ring partition is applied for rotational invariance, and stable statistical rings are generated from these rings to generate hash. The approach proposed in [18] uses the multi-dimensional scaling (MDS) data analysis technique, which learns compact and

discriminative representation.

Feature matrix with log polar transform and discrete Fourier transform is generated from normalized image. In [19], color vector angles are calculated on non-overlapping sub-blocks of an image. The feature matrix is generated by taking DWT on the mean of color vector angle-generated image. The LL components are selected to generate hash. Yan et al. [20] have detected an adaptive feature point by applying SIFT. In the region of detected feature point, the local features are extracted from Stationary Wavelet Transform (SWT).

The main approaches for texture extraction are statistical, structural, filter-based, and model-based. The statistical and filter-based approaches are popular because of their simplicity, computational efficiency, and ample success rate. Local binary operator (LBP) [21] is a statistical-based texture extractor that works in the local and semi-global areas, and became popular due to its ease of use. For a sub-block, the histogram generated with LBP is 256 bin, which is of long length. If LBP is used for image hashing, it breaches the compactness property of hash. Therefore, Center Symmetric Local Binary (CSLBP) [22] is an appropriate choice for hashing. It considers only four cross symmetric neighbours of the pixel in consideration, whereas, LBP considers eight neighbours of the pixel in consideration. CSLBP captures a better structural change using cross-symmetric pair' difference compared to LBP, and also provides rotational invariance. With CSLBP, a 16 bin histogram is generated for a sub-block and results in a shorter length.

In image hashing, texture extraction plays an important role as it gives updates of structural changes in a local area. Basic CSLBP considers only pure sign information. Davarzani et al. [23] have utilizes both sign and magnitude. A histogram is constructed for four magnitudes, each of 16 bin. This results in a total of 64 bin, and the hash size is increased by four times. The magnitude factor that is obtained for every direction does not impact on the discrimination success rate.

The previously proposed approaches [24-27] have used CLSBP as a basic method for texture extraction. In our earlier approaches, the histogram is constructed with the weight factor, which gives a better discrimination power. This has eventually contributed to the success of image hashing algorithm. In the histogram construction with weight factor, the bin increment is based upon the value of weight factor. The considered weight factors are the local characteristics such as the average magnitude difference of cross-symmetric pairs, standard deviation, correlation coefficient, and Laplacian of Gaussian (LoG). The local weight factor captures the local changes that are utilized during the histogram construction. The particular histogram bin is incremented based upon the response of the weight factor. In a plain histogram, bin is always incremented by 1, whereas, a histogram with the weight factor bin increment is directly proportional to the response of weight factor. A further histogram is compressed based on the flipped difference concept. The previously proposed approaches AQ-CSLBP [24], SDQ-CSLBP [25], CoCQ-CSLBP [26], and LoGQ-CSLBP [27] consider the weight factors, such as the average of magnitude difference, standard deviation, correlation coefficient, and Laplacian of Gaussian, respectively. In AQ-CSLBP, CSLBP takes the difference between the cross-symmetric pairs of a pixel. Each pair of difference has a sign and magnitude. Sign is utilized to extract texture features, while the average of magnitude of four cross-symmetric pairs is used as the weight factor. In SDQ-CSLBP approach, the standard deviation of a cross-symmetric pair is used as the weight factor. In CoCQ-CSLBP, the correlation coefficient is measured between the original local image area and the template to determine any structural change. This correlation coefficient is converted into weight during the histogram construction. In the LoGQ-CSLBP approach, the second derivative operator LoG is used, which performs better even in the presence of noise. The LoG operator is calculated on the immediate four neighbors of the center pixel, and their average is taken as the weight factor. In all our previous approaches, after the weighed histogram construction, histogram is compressed by the flipped difference concept [27]. It has been found that, the discrimination power is lower if compression is applied on the histogram that is constructed without any weight factor. Therefore, a short image hash with less discrimination indicates a poor quality. Histogram compression on a histogram constructed by a boosting factor has two advantages. The first advantage is a desirable discrimination power and the second one is the compressed length. On the sender side, the hashing method extracts texture information with local weight factor and the generated histogram incorporates a local change. In transmission, content changes may occur in an image due to various attacks. Hence, on the receiver side, the constructed histogram with weight factor is not

similar to the original one due to the difference in texture as well as local structural information. This dual information, i.e. texture and local structural statistics, achieves an enviable discrimination power.

The rest of this paper is organized as follows: Section 3 presents our novel image hashing technique, using the edge weight factor and the histogram compression technique. Section 4 gives the experimental results of our scheme and some comparable algorithms. We draw our conclusions in Section 5.

3. Proposed method

In the proposed method, the modified LBP and minimum magnitude CSLBP concepts are used. This section explains the pre-processing, modified LBP, minimum magnitude CSLBP, quantization and final hash generation.

3.1. Pre-processing

The proposed method takes into consideration the gray scale images, which are mainly characterized by texture and shape. The input image is kept with a fixed size of 256 × 256 by the bilinear interpolation method. This is done for the experimental purpose and comparison with other methods. In the pre-processing step, the input image is filtered by Gaussian filter to make it robust for content preserving manipulation as well as to reduce the disturbance caused by manipulations like noise, and lossy compression.

3.2. Modified LBP

The modified LBP is applied on the gradient image. In the modified LBP, the input gray scale image is converted to the gradient image by applying a Canny edge detector. The generated gradient image is used by a modified LBP operator to generate an edge weight factor. The LBP extracts image labels from the local neighborhood area of size 3 × 3. An image label represents a local texture that carries the local structural information. This information is significant because whenever any local change takes place on the receiver side, its image label is different from the sender side. Thus the resulting hash has a sufficient discrimination power for content change detection.

In LBP, the pixels in the local block are thresholded by its center pixel value. The thresholded value is either 0 or 1 based on the presence of texture, which is multiplied by powers of two and then added to get an image label for the center pixel, as shown in figure 1(a).

For the 3 × 3 area with eight neighbors, the image label is generated with values from 0 to 255. In the modified LBP, a different order for power of 2 is used, and it is influenced by horizontal, vertical, and diagonal edges, represented in figure 1(b). In hashing, identification of the structural change is important. LBP captures the diagonal difference with respect to the center pixel. This diagonal difference represents the dominant structural change. Therefore, in the modified LBP, more weight is assigned to diagonal locations as diagonal changes are visually more important than the other directions.

1	2	4
128	g_c	8
64	32	16

(a) LBP label

64	8	128
2	g_c	1
32	4	16

(b) Modified LBP label

Figure 1. Labels for local area of 3×3 with center pixel g_c.

$$M-LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \quad (1)$$

$$s(z) = \begin{cases} 1 & z \geq 0 \\ 0 & z < 0 \end{cases} \quad (2)$$

$$W_{M-LBP} = M-LBP_{P,R}(x_c, y_c) \quad (3)$$

where:
z: (g_p-g_c);
g_c: center pixel;
g_p: neighbors of center pixel;
s(z): sign function of LBP;
M-LBP: modified LBP.

$W_{M\text{-}LBP}$ is used as the weight factor during the minimum magnitude CSLBP histogram construction.

3.3. Minimum magnitude CSLBP

CSLBP is used to extract, the rotation invariant texture features in an efficient way. The main purpose behind CSLBP is to generate a small number of labels to produce shorter histograms as well as a high stability for flat image regions. In CSLBP, the difference of only cross-symmetric pairs is taken and thresholded by a non-zero threshold for the robustness on flat areas.

For CSLBP, the local area is chosen as 3 × 3, which is shown in figure 2. P is the number of neighbors, which are 8 and radius R = 1. T is a non-negative value to extract texture for an uneven surface.

CSLBP extracts texture using Equations (4) and (5) [22].

$$CSLBP_{P,R,T}(g_c) = \sum_{p=0}^{P/2-1} s(g_p - g_{p+(P/2)})2^p \quad (4)$$

$$s(z) = \begin{cases} 1 & (z) \geq T \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where:
$z=(g_p-g_{p+(P/2)})$;
T: threshold;
R: radius;
P: no. of neighbors;
g_c: center pixel;
g_p: neighbors of center pixel;
s(z): sign function of CSLBP;
CSBLP: CSLBP texture extractor.

The image label of CSLBP for every pixel lies in the range of 0 - 15, which results in a histogram of 16 bin.

g_5	g_6	g_7
g_4	g_c	g_0
g_3	g_2	g_1

Figure 2. Neighbourhood region with g_c center pixel.

In the proposed method, we compressed a 16 bin CLSBP histogram to a 4 bin one by a minimum magnitude CSLBP. For a pixel, the CSLBP values will be 1, 2, 4, and 8 when g_0, g_1, g_2, and g_3 are greater than theirs opposite pixels g_4, g_5, g_6, and g_7 respectively. The values 1, 2, 4, and 8 indicate the presence of one edge in the lower side. The cross- symmetric pairs are sorted and put in bin 0, i.e. the CSLBP values 1, 2, 4, and 8 are put in the same bin. Before putting into the same bin, their corresponding weight factors are calculated using $W_{Modified\text{-}LBP}$, and are used during the CSLBP histogram construction. Therefore, even if various CSLBP values are put in the same bin for compression, its quality is maintained by its corresponding weight factor. The CSLBP values 3, 5, 6, 9, 10, and 12, which show the presence of two edges in the lower corner of neighborhood are put in the same bin. For these CSLBP values, bin 1 is used. In a similar way, the CSLBP values 7, 11, 13, and 14 that show the presence of three edges are put in bin 2. For the CSLBP values 0 and 15, that represent the exactly the opposite, the texture pair are put in bin 3. In this manner, the 16 bin CSLBP is modified to the 4 bin minimum magnitude CSLBP, and a desirable quality is maintained by the extracting weigh factor.
After CSLBP, a histogram is constructed at a sub-block level. The size of the of the sub-block is an important factor for the success of hashing. For a large sub-block size, the hash size decreases, discrimination and local area forgery detection rate are reduced, and vice versa. The equation for the CSLBP histogram is as follows.

$$H_{MinMag-CSLBP(b)} = \sum_{i=1}^{B}\sum_{j=1}^{B} W_{M-LBP}(i,j) \times f(CSLBP_{MinMag}(i,j), b) \quad (6)$$

$$b \in [0,3]$$

where:
$H_{MinMag\text{-}CSLBP}$: histogram of min. mag. CSLBP;
B:size of sub-block;
b: range of bin for min. mag. CSLBP;
$CSLBP_{Min\text{-}Mag}$: minimum magnitude CSLBP.
All sub-blocks are processed in this manner to generate a 4 bin histogram.

3.5. Quantization and hash generation

A uniform quantization is applied on each sub-block's histogram to generate the binary output. In order to construct the final hash, all block binary outputs are concatenated.
This generates a binary image hash, which is short and has a good discrimination power and short length. On the receiver side, the binary hash can be efficiently compared with the hamming distance. If the hamming distance is less than threshold, then it is content preserving manipulation; otherwise, content change manipulation.

4. Experimental results and analysis

This section discusses the experimental setup, various parameter settings, the results in terms of the hamming distance, and ROC (Receiver Operating Characteristics).

4.1. Setting parameters, and abbreviations for various attacks and comparative methods

The proposed method is experimented by the standard images used in image processing applications. The total test images considered are 36, some of which are taken from Matlab directory and some from the internet. Examples of images are Leena, Baboon, Barbara, and Airplane. These images are analyzed for robustness to content conserving and sensitivity to content change. For this, the test images undergo various malicious and non-malicious operations. A list of various attacks is mentioned in table 1, with parameters and their abbreviations represented in Table 2 for result simplification. The attacks mentioned in table 1 are applied on the test images to generate a modified database. For every

test image, 61 images are generated by applying attacks mentioned in table 1.

Table 1. Various operations with parameter values.

Operations	Descriptions	Parameters
Cropping	Ratio	1%, 3%, 5%, 7%, 9%
Salt & Pepper Noise	Noise Density	0.01, 0.02, 0.03, 0.05, 0.1
Gaussian Noise	Noise Variance	.001, .005, .01, .02, .05
JPEG Compression	Quality Factor	10, 30, 50, 70, 90
Rotate	Rotation Angle	2^0, 4^0, 6^0, 8^0, 10^0
Gamma Correction	Gamma value	0.75, 0.8, 0.9, 1.1, 1.25 4.25, 4.50, 5.00, 5.25
Scaling	Scaling factor	0.7, 0.8, 0.9, 1.1, 1.2 0.01, 0.05, 0.10, 0.15, 0.20
Increase Brightness	Adjustment Range	[0.8 1], [0.6 1], [0.4 1], [0.2 1]
Decrease Brightness	Adjustment Range	[1 0.8], [1 0.6], [1 0.4], [1 0.2]
Increase Contrast	Adjustment Range	[0 0.8], [0 0.6], [0 0.4], [0 0.2]
Decrease Contrast	Adjustment Range	[1 0.8], [1 0.6], [1 0.4], [1 0.2]

The modified database contains a total of 36 × 61 = 2196 images. Based on their content preserving and content change nature, all images are classified into one of the two categories as malicious or non-malicious images. The proposed method is applied on the modified database to generate the hash. The generated binary hash is having a length of 256 bits. Two benchmarks are used to check the performance of the proposed algorithm. One is the hamming distances, which checks the robustness and sensitivity of hash, and the other is the ROC curve, which checks the discrimination capability and determines a successful rate of the proposed algorithm.

For the analysis purpose, the proposed method and the other methods are compared on the basis of the histogram bin, weight factor, and compressed histogram. Based on the histogram bin, the methods are categorized as 256 bin, 16 bin, 64 bin, and 8 bin. Based upon the weight factor, the methods are classified as the average of magnitude, standard deviation, correlation coefficient, and Laplacian of Gaussian. The last category is based on the compressed histogram.

Some of the results obtained are shown in a tabular form, and the others with a graphical representation. For the simplification purpose, in both cases, the methods are denoted with their symbolic names. The method names with their abbreviations and symbolic names are shown in table 3.

Following are the values considered for the different parameters used in experimentation. The size of the input gray scale image is set to 256 × 256 using bilinear interpolation. The low pass Gaussian filter with unit standard deviation is applied on a local area of 3 × 3 for invariant to robustness. For CLSBP computation, R = 1, P = 8, and T = 0.01. CSLBP is calculated for every pixel by fixing a 3 × 3 local region. The CSLBP value for a pixel is between 0 to 3, which results in a 4 bin histogram for a sub-block size of 32 × 32.

Table 2. Symbolic names for various operations

Types of Attack	Symbolic Name	Types of Attack	Symbolic Name
Cropping	A	Salt & Pepper	B
Gaussian	C	Scaling	D
Rotation	E	JPEG	F
Gamma Correction	G	Brightness Plus	H
Brightness Minus	I	Increase Contrast	J
Decrease Contrast	K	Database Average	Avg

Table 3. Symbolic names for methods.

Parameter	Method Name	Symbolic Name	Description
Various bin sizes	LBP [21]	I	256 bin
	CSLBP (Sign) [22]	II	16 bin
	CSLBP (Separate Magnitude) [23]	III	64 bin
	Q-CSLBP [28]	IV	8 bin
Weight factor (16 bin)	A-CSLBP [24]	V	Magnitude Average
	SD-CSLBP [25]	VI	SD
	CoC-CSLBP [26]	VII	CC
	LoG-CSLBP [27]	VIII	LoG
Compressed with weight factor (8 bin)	AQ-CSLBP [24]	IX	Magnitude Average
	SDQ-CSLBP [25]	X	SD
	CoCQ-CSLBP [26]	XI	CC
	LoGQ-CSLBP [27]	XII	LoG

4.2 Robustness and sensitivity analysis by normalized hamming distance

The robustness and sensitivity analyses focus mainly on non-malicious attacks and malicious attacks, respectively. When the image is attacked by non-malicious attacks, the hash code of the original image and its attacked version varies slightly. In this case, the normalized hamming distance between the hash codes of the original and the attacked image is less than the set threshold value. Exactly opposite is the case with the malicious attack, and the hash code of the original image and its attacked version is drastically different.

Depending on the nature of the hash code, a variety of distance metrics are available. For a real value hash, the Euclidean distance is used. For binary values, the Hamming distance is preferable. It is an effective choice for comparison, retrieval, and searching from a large database. The Hamming distance is a simple ex-or operation performed on bitwise fashion between

the hash code of the original image and its attacked version. It is computationally efficient over the Euclidean distance.

$$NHD(H_O, H_A) = \frac{1}{N}\sum_{i=1}^{N}(H_O \oplus H_A) \quad (7)$$

where:
H_O: hash code of original image;
H_A: hash code of modified image;
NHD (H_O, H_A): normalized hamming distance;
N: no. of bits.

The larger the normalized hamming distance, the higher the discrimination. For the proposed method, the normalized hamming distance is about 0.31, which gives the ability to discriminate between content preserving and content change. Tables 4, 5, 6, and 7 show the normalized hamming distance authentic and non-authentic images for methods 1 - 4, 5 - 8, and 9 - 12 and for the proposed method, respectively.

As shown in table 4, method I is implemented with the LBP operator. Method II is implemented with the CSLBP operator, which considers only sign of difference. Method III uses the CSLBP operator with both sign and magnitude. Each magnitude component is represented in a separate histogram. It creates a histogram for each direction, which results in a 64 bin histogram for four directions. Further, because of four histograms sizes, resultant hash size also increases. Method IV implements the CSLBP operator and the compress CSLBP histogram by the flipped difference concept. The results shown in table 4 indicates a poor discrimination for method I and method III.

There should be sufficient NHD between authentic and non-authentic images. The NHD threshold is set for every method. The values of NHD below the set threshold indicates the authentic images, and those having above the set threshold indicates non authentic images.

The results for method I and III clearly show that the NHD values for the set threshold are nearly the same for both the authentic and non-authentic images. This means that, both methods fail to discriminate between the content change and the content modify. All methods mentioned in table 5 use weight factor during CSLBP histogram construction and show a desirable discrimination capability. The results shown for methods mentioned in table 5 use various weight factors without compression. Similarly, all methods mentioned in table 6 are compressed CSLBP ones, having histogram of a 8 bin. With reduced size, the method sustains a desirable distinguishing capability of content change and preserving.

Table 4. NHD for LBP, CSLBP (Sign), CSLBP (Separate Magnitudes) and Q-CSLBP.

Attack	I:LBP		II:CSLBP (Sign)		III:CSLBP (Separate Magnitude)		IV:Q-CSLBP	
	Auth	Non Auth	Auth	Non Auth	Auth	Non Auth	Auth	Non Auth
A	0.00	0.01	0.04	0.1	0.01	0.01	0.05	0.13
B	0.00	0.01	0.03	0.11	0.01	0.01	0.04	0.10
C	0.01	0.02	0.14	0.22	0.01	0.02	0.12	0.19
D	0.00	0.01	0.02	0.14	0.00	0.02	0.02	0.17
E	0.01	0.01	0.05	0.10	0.01	0.01	0.07	0.13
F	0.00	0.01	0.03	0.09	0.00	0.01	0.03	0.10
G	0.00	0.01	0.01	0.12	0.00	0.01	0.01	0.13
H	0.01	0.01	0.04	0.12	0.00	0.01	0.05	0.14
I	0.00	0.01	0.03	0.18	0.00	0.02	0.03	0.21
J	0.01	0.01	0.05	0.17	0.01	0.02	0.06	0.20
K	0.00	0.01	0.04	0.16	0.00	0.02	0.04	0.19

Table 5. NHD for A-CSLBP, SD-CSLBP, CoC-CSLBP, and LoG-CSLBP.

Attack	V:A-CSLBP		VI:SD-CSLBP		VII: CoC-CSLBP		VIII: LoG-CSLBP	
	Auth	Non Auth	Auth	Non Auth	Auth	Non Auth	Auth	Non Auth
A	0.03	0.07	0.04	0.09	0.04	0.10	0.05	0.10
B	0.04	0.08	0.04	0.11	0.03	0.10	0.04	0.11
C	0.06	0.11	0.08	0.14	0.13	0.18	0.13	0.22
D	0.01	0.10	0.01	0.12	0.02	0.14	0.02	0.15
E	0.05	0.08	0.06	0.09	0.06	0.10	0.07	0.11
F	0.01	0.04	0.02	0.05	0.03	0.10	0.03	0.08
G	0.01	0.06	0.01	0.10	0.01	0.12	0.01	0.10
H	0.02	0.06	0.03	0.10	0.04	0.12	0.04	0.10
I	0.01	0.08	0.02	0.17	0.03	0.18	0.02	0.16
J	0.02	0.09	0.03	0.13	0.05	0.17	0.03	0.13
K	0.02	0.09	0.02	0.13	0.04	0.16	0.04	0.15

Table 6. NHD for AQ-CSLBP, SDQ-CSLBP, CoCQ-CSLBP, and LoGQ-CSLBP.

Attack	IX:AQ-CSLBP		X:SDQ-CSLBP		XI: CoCQ-CSLBP		XII: LoGQ-CSLBP	
	Auth	Non Auth	Auth	Non Auth	Auth	Non Auth	Auth	Non Auth
A	0.04	0.11	0.05	0.13	0.05	0.13	0.06	0.14
B	0.06	0.12	0.07	0.15	0.04	0.10	0.05	0.11
C	0.09	0.14	0.11	0.19	0.13	0.17	0.12	0.19
D	0.02	0.17	0.02	0.19	0.02	0.19	0.03	0.19
E	0.07	0.13	0.09	0.15	0.07	0.13	0.08	0.15
F	0.02	0.07	0.02	0.08	0.04	0.13	0.04	0.09
G	0.01	0.14	0.01	0.16	0.02	0.16	0.01	0.12
H	0.05	0.15	0.05	0.17	0.06	0.17	0.04	0.11
I	0.03	0.27	0.03	0.30	0.04	0.26	0.03	0.18
J	0.04	0.18	0.04	0.20	0.08	0.26	0.04	0.15
K	0.04	0.20	0.04	0.21	0.06	0.24	0.04	0.18

However, compression of CSLBP without weight factor leads to a low discrimination capability. Therefore, the desired results are obtained only if the weight factor is used during histogram

construction. Thus for a 16 bin CLSBP, the weight factor is not crucial. However, the weight factor that captures the local essence and binds it into histogram plays an important role for the compressed CSLBP methods.

Table 7. NHD for Proposed Method.

Attack	Proposed Method (T_{NHD}=0.20)	
	Auth.	Non. Auth.
A	0.12	0.23
B	0.11	0.22
C	0.15	0.27
D	0.06	0.29
E	0.19	0.28
F	0.06	0.14
G	0.04	0.25
H	0.08	0.24
I	0.04	0.19
J	0.08	0.28
K	0.08	0.31

4.3 Discrimination capability and success rate of algorithm

A desirable discrimination capability and success rate of algorithm rate in terms of true positive rare (TPR) and false positive rate (FPR) are determined by the ROC curve. The receiver operating characteristic (ROC) is the best benchmark for a binary classifier. The ROC curve is obtained by plotting TPR and FPR on the Y and X axes, respectively, for a method with a particular threshold. Various methods with their thresholds can be plotted simultaneously, and their performance can be observed with the ROC curve [29]. The area under the ROC curve determines the success rate of the algorithm and is proportional it. The area under the ROC curve measures discrimination, i.e. the ability of the test to correctly classify between content preserve and content change. For accurate results, TPR should be high and FPR should be small. Therefore, when a curve is plotted between TPR and FPR, it covers a maximum area. Table 8 shows TPR and FPR for the proposed method for various attacks, and it is followed by a graphical representation, as shown in figure 3.

The ROC curve for the proposed method is plotted in figure 3. This figure shows a clear picture of success for the proposed method. For an average database, TPR is 0.87. For almost all attacks, the proposed method shows a high TPR for a small FPR. The proposed method gives average results only for the JPEG attack as it gives a high TPR for a high FPR.

Table 8, TPR and FPR for Proposed Method.

Attack	Proposed Method	
	TPR	FPR
A	0.74	0.06
B	0.82	0.19
C	0.60	0.04
D	0.99	0.06
E	0.29	0.06
F	0.99	0.58
G	1.00	0.09
H	0.89	0.03
I	1.00	0.25
J	0.89	0.10
K	0.92	0.17
Avg.	0.87	0.11

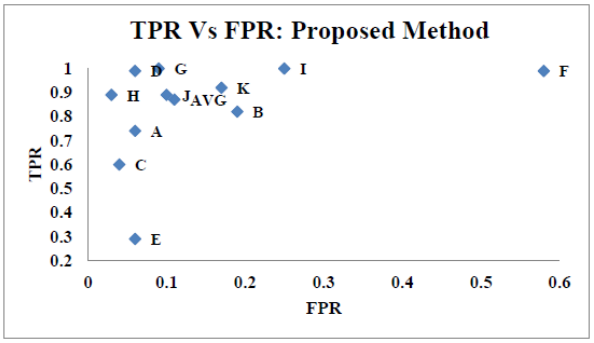

Figure 3. TPR *vs.*FPR for proposed for various attacks.

Instead of a tabular representation, the results obtained for ROC are graphically represented for some attacks. Figure 4 - 14 show the ROC curves for various attacks. The ROC figures clearly show that results of methods I, and III have a poor discrimination power as TPR is low for a relatively high FPR. Methods II, V, VI, VII, and VIII are all 16 bin and have a better discrimination power. Methods IV, IX, X, XI, and XII are all 8 bin. Method IV has a poor discrimination power compared to methods IX, X, XI, and XII. Method IV generates a plane histogram, while the rest of the four methods generate a histogram with a weight factor. The performance of the proposed method is equally comparable to 8 bin and 16 bin methods.

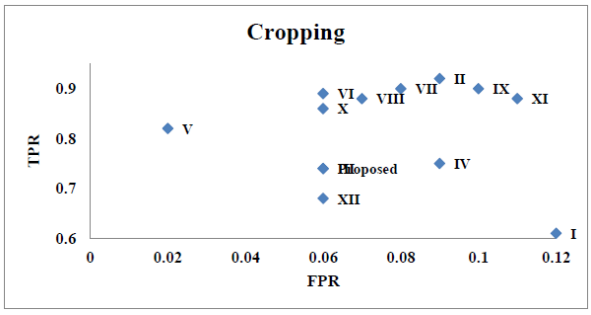

Figure 4. TPR *vs.* FPR: Cropping.

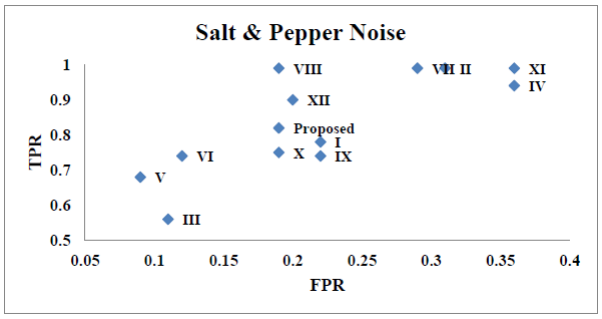

Figure 5. TPR *vs.* FPR: Salt & Pepper Noise.

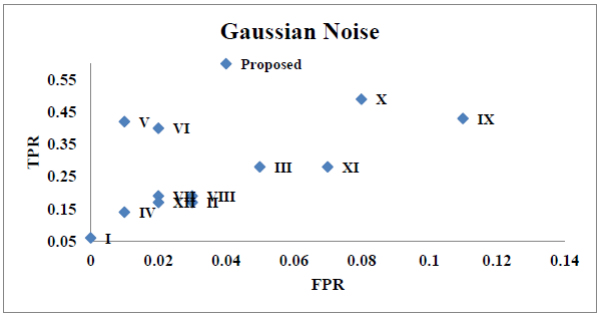

Figure 6. TPR *vs.* FPR: Gaussian Noise.

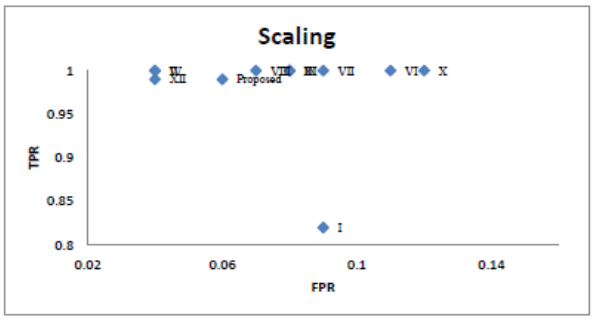

Figure 7. TPR *vs.* FPR: Scaling.

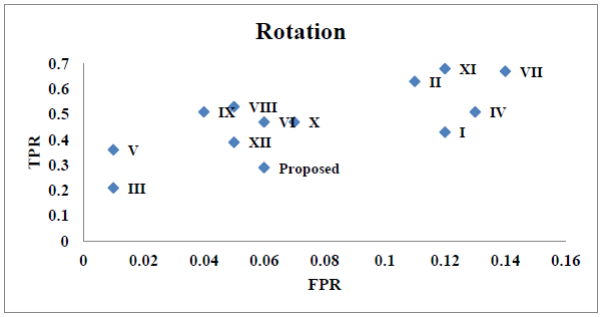

Figure 8. TPR *vs.* FPR: Rotation.

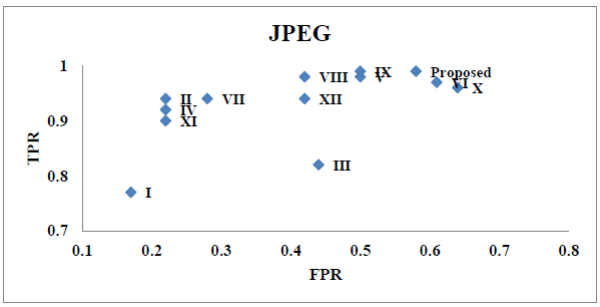

Figure 9. TPR *vs.* FPR: JPEG.

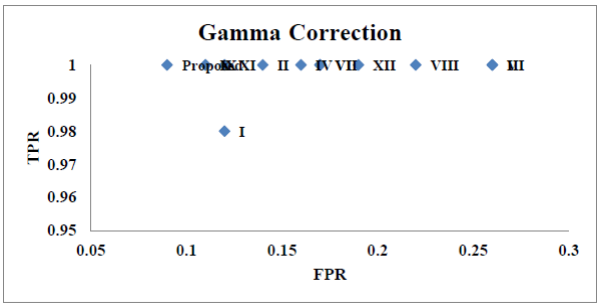

Figure 10. TPR *vs.* FPR: Gamma Correction.

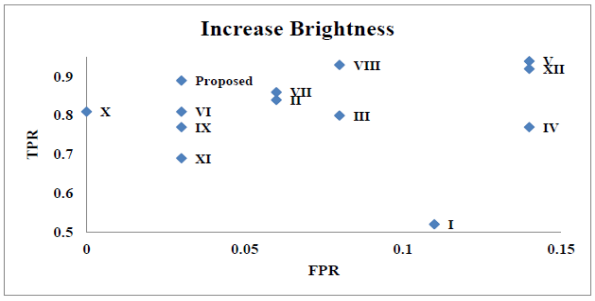

Figure 11. TPR *vs.* FPR: Increase Brightness.

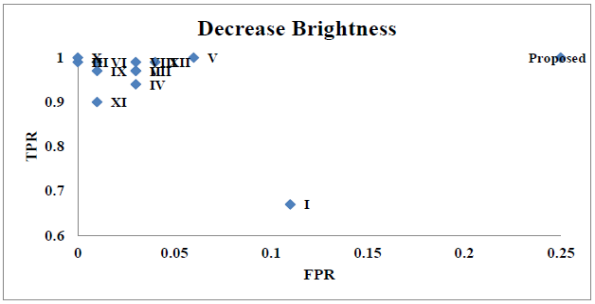

Figure 12. TPR *vs.* FPR: Decrease Brightness.

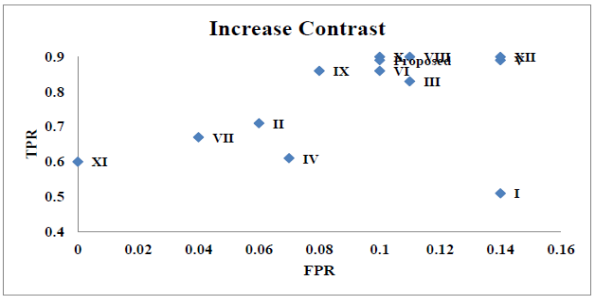

Figure 13. TPR *vs.* FPR: Increase Contrast.

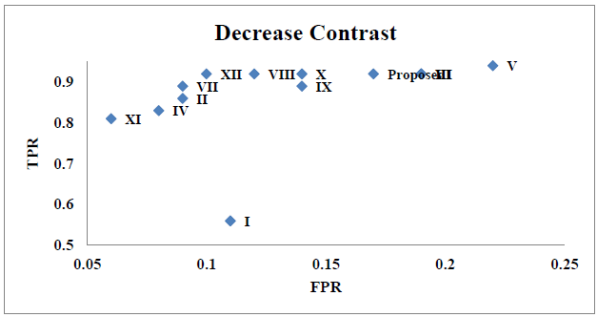

Figure 14. TPR *vs.* FPR: Decrease Contrast.

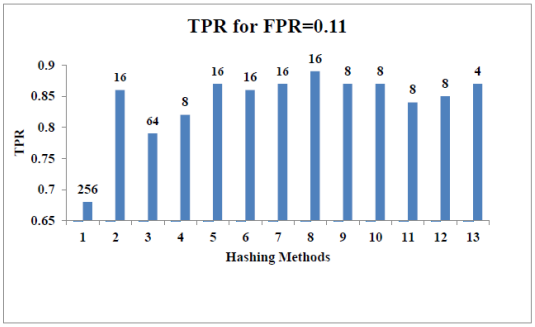

Figure. 15. TPR and Bin No. for FPR = 0.11 for various Methods.

Figure 15 shows the performance of all methods (methods I to XII and the proposed method) in terms of the histogram bin for sub-block and TPR for a fixed value of FPR = 0.11.

5. Conclusion

We have proposed a compressed image hashing method with a weight factor and a compressed CSLBP histogram. Unlike the existing hashing methods, the proposed method takes full advantage of the local changes incorporated as the weight factor during the CSLBP histogram construction and histogram compression by utilizing the texture distribution of CSLBP. Local changes are extracted from the gradient input image by the modified LBP label. The 16 bin CSLBP histogram is compressed to 4 bin without comprising quality. The compact length and desirable discrimination power are two main characteristics of hashing achieved by the proposed method. The proposed method is robust to a variety of attacks as the results are proved by the NHD and ROC curves.

References

[1] Sun, R., Zeng, W. and Yan, X. (2011). Perceptual image hashing method using Contourlet HMT Model, 3rd IEEE International Conference on Multimedia Information Networking and Security (MINES), Shanghai, China, 2011.

[2] Srivastava, M., Siddiqui, J. and Ali, M.A. (2016). Robust image hashing based on statistical features for copy detection, IEEE International Conference on Electrical, Computer and Electronics Engineering (UPCON), Varanasi, India, 2016.

[3] Kang, L.W., Lu, C.S. and Hsu, C.Y. (2009). Compressive sensing-based image hashing, 16th IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, 2009.

[4] Mo, Z.W., Zhu, Y.S. and Liu, Z. (2014). A hash-based image content authentication scheme for tamper detection using compressive sensing, 11th IEEE International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 2014.

[5] Lei, Y.Q., Chau, K.Y., Lu, Z.M. and Ip, W.H. (2010). DCT-domain global feature and DWT-domain least-squares line fitting based local feature for robust image hashing, International Journal of Innovative Computing, Information and Control, vol. 6, no. 6, pp. 2513-2521.

[6] Guo, X.C. and Hatzinakos, D. (2007). Content based image hashing via wavelet and radon transform, Springer Pacific-Rim Conference on Multimedia, Berlin, Heidelberg, 2007.

[7] Yu, F.X., Lei, Y.Q., Wang, Y.G. and Lu, Z.M. (2010). Robust image hashing based on statistical invariance of dct coefficients, Journal of Information Hiding and Multimedia Signal Processing, vol. 1, no. 4, pp. 286-291.

[8] Karsh, R.K., Laskar, R.H. and Richhariya, B.B. (2016). Robust image hashing using ring partition-PGNMF and local features, SpringerPlus, vol. 5, no.1, pp.1995.

[9] Zhao, Y., Wang, S., Zhang, X. and Yao, H. (2013). Robust hashing for image authentication using Zernike moments and local features, IEEE Transactions on Information Forensics and Security, vol. 8, no. 1, pp.55-63.

[10] Soman, G. and John, J.K. (2016). Block-based forgery detection using global and local features, Springer International Conference on Soft Computing Systems, New Delhi, India, 2016.

[11] Sebastian, L.S., Varghese, A. and Manesh, T. (2015). Image authentication by content preserving robust image hashing using local and global features, Elsevier International Conference on Information and Communication Technologies, ICICT, Kochi, India, 2015.

[12] Wang, X., Pang, K., Zhou, X., Zhou, Y., Li, L. and Xue, J. (2015). A visual model-based perceptual

image hash for content authentication, IEEE Transactions on Information Forensics and Security, vol. 10, no. 7, pp.1336-1349.

[13] Lai, H., Yan, P., Shu, X., Wei, Y. and Yan, S. (2016). Instance-aware hashing for multi-label image retrieval, IEEE Transactions on Image Processing, vol. 25, no. 6, pp.2469-2479.

[14] Lv, X. and Wang, Z.J. (2012). Perceptual image hashing based on shape contexts and local feature points, IEEE Transactions on Information Forensics and Security, vol. 7, no. 3, pp.1081-1093.

[15] Tang, Z., Zhang, X. and Zhang, S. (2014). Robust perceptual image hashing based on ring partition and NMF, IEEE Transactions on knowledge and data engineering, vol. 26, no. 3, pp.711-724.

[16] Tang, Z., Zhang, X., Li, X. and Zhang, S. (2016). Robust image hashing with ring partition and invariant vector distance, IEEE Transactions on Information Forensics and Security, vol. 11, no. 1, pp.200-214.

[17] Tang, Z., Zhang, X., Huang, L. and Dai, Y. (2013). Robust image hashing using ring-based entropies. Signal Processing, vol. 93, no.7, pp. 2061-2069.

[18] Tang, Z., Huang, Z., Zhang, X. and Lao, H. (2017). Robust image hashing with multidimensional scaling. Signal Processing, 137, pp.240-250.

[19] Tang, Z., Dai, Y., Zhang, X., Huang, L. and Yang, F. (2013). Robust image hashing via colour vector angles and discrete wavelet transform. IET Image Processing, vol. 8, no. 3, pp.142-149.

[20] Yan, C.P., Pun, C.M. and Yuan, X.C. (2016). Multi-scale image hashing using adaptive local feature extraction for robust tampering detection, Signal Processing, vol. 121, pp.1-16.

[21] 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, vol. 24, no. 7, pp. 971-987.

[22] Xiao, J. and Wu, G. (2011). A robust and compact descriptor based on center-symmetric LBP, IEEE 6th International Conference on Image and Graphics (ICIG), Anhui, China, 2011.

[23] Davarzani, R., Mozaffari, S. and Yaghmaie, K. (2015). Image authentication using LBP-based perceptual image hashing, Journal of AI and Data Mining, vo.3 no. 1, pp. 21-30.

[24] Patil, V. and Sarode, T. (2016). Image hashing using AQ-CSLBP with double bit quantization, IEEE International Conference on Optoelectronics and Image Processing (ICOIP), Warsaw, Poland, 2016.

[25] Patil, V. and Sarode, T. (2016). Image hashing by SDQ-CSLBP, IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, India, 2016.

[26] Patil, V. and Sarode, T. (2016). Image hashing by CCQ-CSLBP, IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), Pune, India, 2016.

[27] Patil, V. and Sarode, T. (2016). Image hashing by LoG-QCSLBP, ACM 2nd International Conference on Communication and Information Processing, Singapore, 2016.

[28] Baber, J., Satoh, S.I., Afzulpurkar, N. and Bakhtyar, M. (2012). Q-CSLBP: compression of CSLBP descriptor, Springer Pacific-Rim Conference on Multimedia, Berlin, Heidelberg, 2012.

[29] Fawcett, T. (2006). An introduction to ROC analysis, Pattern Recognition Letters, vol. 27, no. 8, pp. 861-874.

نشریه هوش مصنوعی و داده کاوی

درهم سازی فشرده تصویر با استفاده از مقدار کمینه‌ی الگوی دودویی محلی متقارن مرکزی

Varsha Santosh Patil* و **Tanuja K Sarode**

دانشکده مهندسی کامپیوتر، دانشکده فنی و مهندسی تادومال شاهانی، دانشگاه بمبئی، بمبئی، هند.

ارسال ۲۰۱۸/۰۱/۲۶؛ بازنگری ۲۰۱۸/۰۲/۲۳؛ پذیرش ۲۰۱۸/۰۴/۰۶

چکیده:

تکنیک درهم‌سازی تصویر پردازش‌هایی چون فشرده‌سازی و بهسازی را بر روی تصاویر دیجیتال مجاز می‌داند. به عبارت دیگر، با انجام این عملیات بر روی یک تصویر کد درهم سازی آن تغییر نخواهد کرد. توابع درهم‌سازی در رمزنگاری، به تغییرات مقادیر بیت‌های تصویر بسیار حساس هستند. عملیات درهم‌سازی تصویر، مجموعه‌ای از ویژگی‌های مهم موجود در تصویر را به صورت عددی ارائه میدهد. از این‌رو در این مقاله، یک الگوریتم جدید درهم‌سازی تصویر به‌منظور احراز هویت ارائه می‌شود که در برابر انواع حملات مقاوم است. در رویکرد پیشنهادی، کد درهم‌سازی کوتاهی با استفاده از مقدار کمینه‌ی الگوی دودویی محلی متقارن مرکزی (CSLBP)، بدست می‌آید. در این روش، درهم سازی تصویر با استفاده از فاکتور وزن لبه استخراج شده به کمک یک نسخه بهبود یافته از تکنیک الگوی دودویی محلی بر روی گرادیان تصویر، قادر است تصاویر مختلف را از هم تمییز دهد. به عبارت دیگر، در روش پیشنهادی ویژگی‌های بافت تصویر توسط روش CSLBP استخراج می‌شود. قدرت درهم ساز تصویر در تمایز بین تصاویر مختلف با بهره گیری از هیستوگرام CSLBP فاکتور وزن لبه ها افزایش می‌یابد. با بهره گیری CSLBP اندازه هیستوگرام ایجاد شده، به یک چهارم هیستوگرام اصلی کاهش می‌یابد. در واقع کد درهم‌سازی کوتاه و امکان تمایز قابل قبول بین تصاویر درهم‌سازی شده، دو مزیت روش پیشنهادی هستند. ارزیابی نتایج تجربی بر اساس معیارهای فاصله همینگ، نرخ مثبت صحیح، نرخ مثبت کاذب و منحنی مشخصه ROC، نشان دهنده عملکرد مناسب روش پیشنهادی است.

کلمات کلیدی: احراز هویت، الگوی دودویی محلی متقارن مرکزی (CSLBP) ، الگوی دودویی محلی، درهم سازی، هیستوگرام..