

# Image Hashing Using DWT-CSLBP

Varsha Patil\*, Tanuja Sarode

TSEC, Univerity of Mumbai, Mumbai, India.

\* Corresponding author. Tel: 919768287595; email: varshasp2977@gmail.com

Manuscript submitted January 7, 2019; accepted March 8, 2019.

doi: 10.17706/jcp.14.3.210-222

---

**Abstract:** Center Symmetric Local Binary Pattern (CSLBP) is widely used in texture and object detection, but its utilization in image hashing is still limited. Image hashing is a powerful technique to identify whether image content is changed or preserved. Two main issues in image hashing are its compact length and its robust discrimination power. To deal with these issues, a robust image hashing method is proposed by combining features of CSLBP and Discrete Wavelet Transform(DWT). CSLBP is applied on LL band of DWT for compact length hash. Discrimination power of hashing is enhanced by weigh factor which comprises information from all sub bands of 1-Level DWT. All sub-band images of 1-Level DWT is combined to form secondary image. Local statistical features like Standard Deviation and Magnitude Average are calculated to obtain local weight factor MASD (Magnitude Average and Standard Deviation). Weighted LL-CSLBP histogram is constructed using this weight factor. Experimental results are demonstrated with Receiver Operating Characteristics (ROC) and Normalized Hamming Distance (NHD). The results shows that, the proposed scheme is robust against content preserving operations and sensitive to malicious operations.

**Key words:** Authentication, image hashing, discrete wavelet transform (DWT), center symmetric local binary pattern (CSLBP), double bit quantization (DBQ).

---

## 1. Introduction

With the prevalent use of internet, data protection has become an important issue. Malicious data manipulation has serious consequences in various sensitive fields like legal matters, scientific research, forensic investigations and government documents [1], [2].

After capturing multimedia data like image and video, post processing operations on multimedia data are common. These operations are generally subject oriented like gamma correction, contrast change or JPEG compression etc. to improve the quality of the image. The purpose behind post processing is twofold, unintentional and intentional. Unintentional operations preserve information content, only allows change in appearance. Intentional lead to malicious operations which results in original content change. The idea behind image hashing is to identify content preservation. Image hashing is convenient for multimedia data. For text data authentication, Cryptographic hashing is suitable option which plays important role in applications like data integrity, password authentication etc [3]. Multimedia data is generally prone to changes. Image hashing allows free changes to content preserving and confines to original content change.

Image hashing algorithm should satisfy the following important properties: (i) Perceptual sensitivity/robustness: Hash algorithm should be robust to content preservation (visually akin) and sensitive to content change. (ii) Discerning competence: Similar images have similar hash value and vice

versa. iii) Uniqueness: Hash code should be unique, so there would not be a collision between two different images. iv) Compact size and efficient search: As the resultant hash code is stored in the image header, compactness becomes an important concern. Image hash code is stored in binary, so that efficient and fast comparison could be possible on the receiver side [4].

In our proposed approach, main focus is on the texture features extraction by CLSBP. Along with texture features, information from the different bands of wavelet is used to generate weight factor. During histogram generation of LL-CSLBP, this weight factor is utilized to enhance discrimination power of image hash.

## **2. Review of Literature**

We reviewed various approaches of perceptual hashing algorithms, based on image features.

Global and local feature combined approach: Projected gradient non-negative matrix factorization (PGNMF) is applied on ring partition to generate rotation invariant global features. while local features are extracted by saliency object detection [5]. Global features are obtained by Zernike moments from luminance and chrominance component of an image. For local feature, saliency map is used which obtains texture information as well as position of significant regions of an image [6]. In similar approach, global features are retrieved from Zernike moments and local features are acquired from Haralick textures [7]. Haralick texture features and MOD-LBP with Zernike moments are combined to form a hash [8].

Transformed Domain: To generate a hash, concept of Multidimensional Scaling (MDS) is used which differentiates compact and discriminative representation from feature matrix [9]. Translation and scaling invariant moments are calculated from Radon Transformed (RT) image. For rotation robustness, Discrete Fourier Transform (DFT) features are obtained from calculated moments. Significant DFT coefficients are selected, normalized and quantized to generate hash [10]. Fourier transformed image is converted into polar coordinates. Quantization is applied to generate a binary hash [11]. 2-Level DWT is generated from  $L^*a*b$  of input image. Sequence of circles on approximation and feature is applied  $LL_2$  band for hash generation [12]. Invariance property of the Fourier Mellin Transform (FMT) is captured to generate hash [13]. In similar approach, Log-Polar transform, which is a part of the Fourier-Mellin transform is utilized for hash generation [14].

Matrix Factorization: SVD, NMF, PCA etc. are matrix factorization based techniques. Secondary image is obtained by applying Non-negative Matrix Factorization (NMF) on a pseudo random sub-images generated from original image. NMF again applied on the secondary image to form a NMF-NMF vector. Hash bits are generated from this generated vector [15]. Low pass filtering is used on luminance part of the image to generate a secondary image. Invariant features are generated by applying NMF on the secondary image. Invariant features are quantized to generate hash [16]. Further to overcome the problem of rotation attack, the secondary image is partitioned using rings of different radius [17]. Compressed Sensing (CS) process is used for tamper detection and localization. CS maintains all information of sub blocks. For short length hash, Singular Value Decomposition (SVD) is used to select components having most energy [18]. Singular Value Decomposition (SVD) is used to find a suitable and efficient representation of the image from which the correlation coefficients are calculated to generate a hash [19].

Moment and GLCM: Invariant Hu's moments are extracted from YCbCr and HSI colour space. Resultant hash is concatenation of invariant moments of both color components [20]. Discrete Cosine Transformation (DCT) and local variations in the Gray Level Co-occurrence Matrix (GLCM) are used on several rings of the input image to generate rotation invariant hash [21]. Tchebichef moments represents the image under orthogonal kernel having qualities of orthogonality and robustness which can identify content change attacks. These invariant features of Tchebichef moments are used to generate hash using adaptive

quantization [22].

CSLBP and its variants: Image hashing using LBP [23] is not a feasible option, because of a long histogram, which increases hash size and decreases resultant discrimination power of hash descriptor. CSLBP [24] is the compressed version of LBP with desirable properties like rotational invariance, computational efficiency, short histogram. CSLBP is essentially developed as the local region descriptor. The purpose behind CSLBP is to generate small number of LBP labels for shorter histograms. Unlike LBP, in CSLBP, pixel values are not compared to the center pixel. Instead of center pixel comparison, opposite pixel pairs are compared. CSLBP had taken into account only sign information.

Davarzani *et al.* [25] has attempted to use sign as well as magnitude (gradient) of cross pair differences. This approach generates histogram of 16 bin for each direction. It results into 64 bin histogram. Therefore the method is not suitable due to increased hash size. Baber *et al.* [26] compressed 16 bin CSLBP histogram into 8 bin histogram by using flipped difference concept. It has been found that flipped difference concept compresses histogram at the cost of reduced discrimination power. To overcome this problem, in our previous approaches, we used various statistical weight factor during CSLBP histogram construction to retain discrimination power. In AQ-CSLBP [27], weighted CSLBP histogram is constructed. Average of magnitude of four cross symmetric pairs of local region is used as a weight factor. Average of magnitude represent local gradient information. Thus sign as well as local measures are used in previously proposed weighted CSLBP approaches. 16 bin CSLBP histogram is compressed using flipped difference to 8 bin. Further in similar approaches, like SDQ-CSLBP [28], standard deviation of four cross symmetric neighbours is treated as a weight factor. In CoCQ-CSLBP [29], correlation coefficient, between template and local image region is calculated and converted to weight factor. LoGQ-CSLBP [30] makes the use of Laplacian of Gaussian (LoG) as a weight factor. LoG is a second order derivative edge detection operator. Because of Gaussian smoothing, it extracts edges well even in the presence of noise.

The rest of the paper is organized as follows: Proposed method described in Section III. Section IV gives experimental results and analysis. Conclusions drawn are in Section V.

### 3. Proposed Method DWT-CSLBP Image Hashing

#### 3.1. Pre-processing

Using bilinear interpolation method, size of the input gray scale image is set to  $256 \times 256$  for experimental analysis and comparison. Input image is smoothen by Gaussian filter. Smoothen input image is robust to adverse effect caused by manipulations like noise and lossy compression etc.

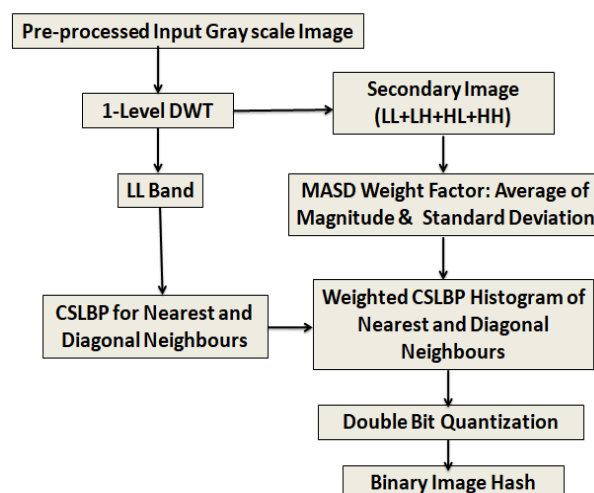


Fig. 1. Block diagram of DWT-CSLBP.

Discrete wavelet transform, is applied on a pre-processed image to divide it into four parts. Wavelet transform is a convolution operation, which uses both high pass and low pass filtering in the vertical and horizontal direction. A single level 2D-DWT splits image into four quadrants. Each quadrant represents particular frequency sub-band image. LL, LH, HL and HH are four quadrants. LL band stores significant coefficients which characterizes coarse information. LL band represents average (low frequency component) information which visually looks same as original image but in reduced size. Remaining three frequency bands LH, HL and HH represents vertical, horizontal and diagonal edge information.

All sub-band images LL, LH, HL and HH are combined to generate secondary image as shown in equation (1). MASD weight factor is formed by local statistical features such as Standard Deviation and Magnitude Average. These features are obtained from local region of secondary image. This weight factor is used in construction of weighted CSLBP histogram.

$$SI = (LL^2 + LH^2 + HL^2 + HH^2) \tag{1}$$

where *SI* represents secondary image; *LL*, *LH*, *HL* and *HH* represent four sub-bands of one level DWT

### 3.2. MASD ( Magnitude Average + Standard Deviation) Weight Factor

MASD local weight factor is combination of two statistical measures namely Magnitude Average and Standard Deviation. Gradient information in the form of magnitude difference of four cross-symmetric pairs is calculated and their average is taken. Magnitude Average represents average gradient information that is local region edge details. Standard Deviation shows how much variation is there from the average. Unlike mean, Standard deviation is robust measure. As Standard Deviation (SD) measures variability, it can represents edge variation when value of SD is more. Following equations represents Magnitude Average, Standard Deviation and MASD weight factor formulation respectively.

SI <sub>5</sub>	SI <sub>6</sub>	SI <sub>7</sub>
SI <sub>4</sub>	SI <sub>c</sub>	SI <sub>0</sub>
SI <sub>3</sub>	SI <sub>2</sub>	SI <sub>1</sub>

Fig. 2. Local region of secondary image.

$$\begin{aligned} Mag_{0,4} &= |SI_0 - SI_4| & Mag_{1,5} &= |SI_1 - SI_5| \\ Mag_{2,6} &= |SI_2 - SI_6| & Mag_{3,7} &= |SI_3 - SI_7| \end{aligned} \tag{2}$$

$$MA = \frac{1}{4} [Mag_{0,4} + Mag_{1,5} + Mag_{2,6} + Mag_{3,7}] \tag{3}$$

$$SD = \sqrt{\frac{\sum (SI - \overline{SI})^2}{9}} \tag{4}$$

where *SI* represents local area and  $\overline{SI}$  represents mean of observations

### 3.3. Weighted CSLBP

CSLBP is calculated over *LL* band. For CLSBP computation, parameters are set to radius *R* = 1, neighbourhood *P* = 8, and *T* = 0.1. Center-symmetric local binary patterns (CSLBP) for a local region of 3×3

is defined as shown in equation (5) and (6).

$$CSLBP_{P,R,T}(g_c) = \sum_{p=0}^{P/2-1} \text{sign}(g_p - g_q) 2^p \quad (5)$$

$$\text{sign}(g_p - g_q) = \begin{cases} 1, & (g_p - g_q) \geq T \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

where  $q$  is  $p+(P/2)$ ;  $T$  represents threshold to extract texture;  $P$  represents neighbourhood pixels;  $R$  represents radius and determined number of neighbours;  $\text{sign}(g_p-g_q)$  is the thresholding criteria.

Gray-level difference between cross-symmetric pairs around a center pixel is compared against threshold  $T$ .  $T$  helps to extract texture feature on flat image region. With cross-symmetric pairs,  $CSLBP$  confines superior gradient details. Equation (5) clearly demonstrate that for  $CSLBP$  the value of each pixel varies between 0 to 15. To generate compact hash,  $CSLBP$  equation is not applied on all four cross symmetric pairs simultaneously. Instead it is applied based on position of neighbours. Neighbourhood region is divided into two types one is nearest and the other is diagonal neighbours. Nearest neighbours pairs are  $g_0$  &  $g_4$  and  $g_2$  &  $g_6$ . Diagonal neighbours pairs are  $g_1$  &  $g_5$  and  $g_3$  &  $g_7$ . Based on classification of neighbours, equation (5) is modified as below.

$$CSLBP_{NN} = s(g_0 - g_4) \times 2^0 + s(g_2 - g_6) \times 2^1 \quad (7)$$

$$CSLBP_{DN} = s(g_1 - g_5) \times 2^0 + s(g_3 - g_7) \times 2^1 \quad (8)$$

From equation (7) and (8), values of  $CSLBP$  lies between 0 to 3 for each pixel.

After  $CSLBP$  calculation for different neighbours, two histograms of 4 bin are constructed based on neighbours. For histogram, size of the sub-block is set to  $32 \times 32$ . To increase discrimination quality of hash, weight are used during  $CLSBP$  histogram construction. For nearest neighbours  $MA$  (Magnitude of Average) weight factor is used while for diagonal neighbours  $SD$  (Standard Deviation) weight factor is used as shown in following equation (9) and (10).

$$H_{DWT-CSLBP_{NN}}(b) = \sum_{i=1}^B \sum_{j=1}^B MA(i, j) \times f(CSLBP_{NN}(i, j), b) \quad (9)$$

$$H_{DWT-CSLBP_{DN}}(b) = \sum_{i=1}^B \sum_{j=1}^B SD(i, j) \times f(CSLBP_{DN}(i, j), b) \quad (10)$$

where  $B$  represents size of the sub-block.  $CSLBP_{NN}(i, j)$  and  $CSLBP_{DN}(i, j)$  is  $CSLBP$  value for pixel,  $f$  is bin increment function,  $MA(i, j)$  is Magnitude Average and  $SD(i, j)$  Standard Deviation weight factor for a pixel.

After histogram construction, double bit quantization is performed on each histogram of the sub block, to generate image hash code.

Double bit quantization: Weighted  $CSLBP$  extract texture and edge features in block-wise manner. These features are in real values. These features are double bit quantized to store in simplified form with minimum loss. Quantization is lossy step. However weighted  $CSLBP$  is applied only on LL sub-band image. Therefore extracted features are small in size and there should be minimum loss during quantization. To achieve this, like our pervious proposed approaches, histogram is not compressed and double bit quantization (DBQ) is preferred over single bit quantization (SBQ) to produce quality hash code. Following equation (11) and (12) represent equation for double bit quantization.

$$b = \frac{h_{max} - h_{min}}{2} \quad (11)$$

where  $h_{max}$  and  $h_{min}$  represents lowest and highest of the frequencies of a histogram.

$$\text{Quantized Threshold} = \text{round}((\text{Original value} - h_{min}) / b) \quad (12)$$

Quantized Threshold represents threshold. It quantized histogram values into three levels 0, 1 and 2. All sub-blocks weighted histogram is passed through double bit quantization to generate values in the range from 0 to 2.

In the proposed *DWT-CSLBP*, all band information is represented in compact manner by secondary image. *LL-CSLBP* is used to generate short length hash code. To enhance discrimination power of an image hash, two weight factors which extract local edge details are combined. To retain the quality of extracted features and to represent in simplified form, double bit quantization is used.

#### 4. Experimental Results and Analysis

In the experiment, all images are resized to  $256 \times 256$  by using bilinear interpolation and normalized by applying gaussian low pass filter with unit standard deviation. For CSLBP calculation  $R = 1$ ,  $P = 8$  and  $T = 0$ . 1. Single level 2-D DWT (Haar) is applied to produce four bands. For histogram generation, sub block size is set to  $32 \times 32$ . For input image, size of each band is  $128 \times 128$ . Number of sub blocks for LL band is 16. To maintain quality of image hash, double bit quantization is used with three levels. Image hash size = (No. of sub block)  $\times$  (No. of Histogram bin)  $\times$  (No. of encoding bits). Therefore, for 8 bin, the minimum size of hash code is  $16 \times 8 \times 2 = 256$  bits.

Total 36 standard images are taken from Matlab directory and the interne. Most commonly used image manipulations are taken into account. Some operations are malicious and some are non-malicious. For example, for any attack, if image content is preserved, image is categorized as a non-malicious otherwise it is categorized as a malicious. It is not practical to show results of each image separately. Therefore, for every attack, average of all results is presented.

Table 1 lists the operations with parameters applied on each image to generate set of test images. Abbreviations used for various operations and methods are given in Table 2 and 3.

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	0.001, 0.005, 0.01, 0.02, 0.05
JPEG Compression	Quality Factor	10, 30, 50, 70, 90
Rotate	Rotation Angle	20, 40, 60, 80, 100
Gamma Correction	Gamma value	0.75, 0.8, 0.9, 1.1, 1.25, 4.25, 4.50, 4.75, 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	Range of adjustment	[0.8 1],[0.6 1],[0.4 1],[0.2 1]
Decrease Brightness	Range of adjustment	[0 0.6],[0 0.4],[0 0.2],[0 0.1]
Increase Contrast	Range of adjustment	[0 0.8], [0 0.6], [0 0.4], [0 0.2]
Decrease Contrast	Range of adjustment	[0.8 1], [0.6 1], [0.4 1], [0.2 1]

Image hash is calculated of all input images and their attacked versions for experimental evaluation. Proposed method is mainly tested for its robustness for content preserving and sensitivity to content change. It is evaluated by two standard benchmark, one is Normalized Hamming Distance (NHD) and other is ROC. NHD result shows robustness to content preserving and sensitive to content change. ROC characteristics determines success rate and hence discrimination capability of hashing algorithm.

Table 2. Abbreviations for Various Operations

Types of Attack	Abbreviations	Types of Attack	Abbreviations
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	Average of Database	Avg

Table 3. Abbreviations for Existing and Proposed Hashing Methods (8 bin)

Method	Method Name	Abbreviations
Existing hashing methods	LBP (256 bin) [23]	E1
	CSLBP (16 bin) [24]	E2
	CSLBP (Sep. Mag.) (64 bin) [25]	E3
	Q-CSLBP (8 bin) [26]	E4
Proposed hashing Methods (16 bin)	AQ-CSLBP [27]	PQ1
	SDQ-CSLBP [28]	PQ2
	CCQ-CSLBP [29]	PQ3
	LoGQ-CSLBP [30]	PQ4
	DWT-CSLBP	PQ5

Normalized Hamming Distance: Normalized Hamming Distance is used to measure similarity between two image hash codes. Hamming distance represents simply XOR operation. Number of error bits is equal to hamming distance.

Receiver Operation Characteristics (ROC): True positive rate (TPR) and false positive rate (FPR) are important entities of any binary classifier system for performance measurement. ROC curve represents TPR on y axis and FPR on x axis for various methods with their individual threshold. Higher TPR indicates stronger robustness for same FPR while FPR indicates better discriminative capability for same TPR.

#### 4.1. Results of DWT-CSLBP Image Hashing

Table 4 and Table 5 shows results of NHD and ROC for DWT-CSLBP. Their corresponding graphical representation is shown in Fig. 3 and Fig. 4 respectively.

Observations: For DWT-CSLBP,  $T_{NHD}$  is set to 0.20.  $T_{NHD}$  condition is satisfied for almost all types attacks except JPEG non-authentic images.

Table 4. NHD for *DWT-CSLBP*

Attack	<i>DWT-CSLBP</i> [ $T_{NHD}=0.20$ ]	
	Auth	Non Auth
Cropping	0.10	0.24
Salt & Pepper Noise	0.13	0.26
Gaussian Noise	0.14	0.29
Scaling	0.04	0.37
Rotate	0.20	0.31
JPEG Compression	0.04	0.09
Gamma Correction	0.03	0.28
Increase Brightness	0.09	0.26
Decrease Brightness	0.03	0.32
Increase Contrast	0.08	0.38
Decrease Contrast	0.08	0.38

Table 5. TPR and FPR for *DWT-CSLBP*

Attack	<i>DWT-CSLBP</i>	
	TPR	FPR
Cropping	0.86	0.07
Salt & Pepper Noise	0.68	0.20
Gaussian Noise	0.64	0.11
Scaling	1.00	0.08
Rotate	0.22	0.00
JPEG Compression	1.00	0.94
Gamma Correction	1.00	0.05
Increase Brightness	0.90	0.00
Decrease Brightness	1.00	0.00
Increase Contrast	0.85	0.06
Decrease Contrast	0.92	0.18
Avg. Database	0.87	0.11

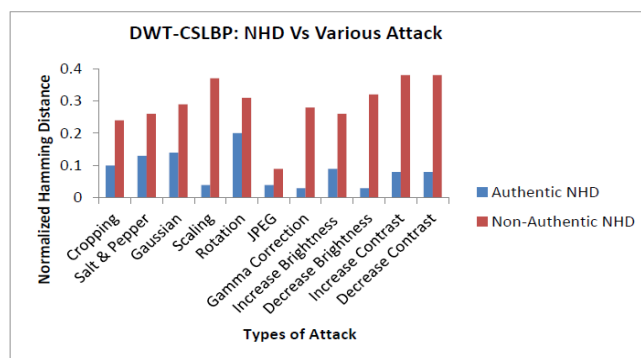


Fig. 3. NHD for DWT.



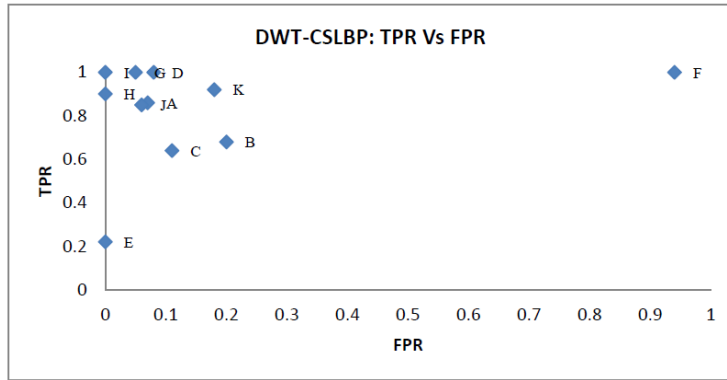
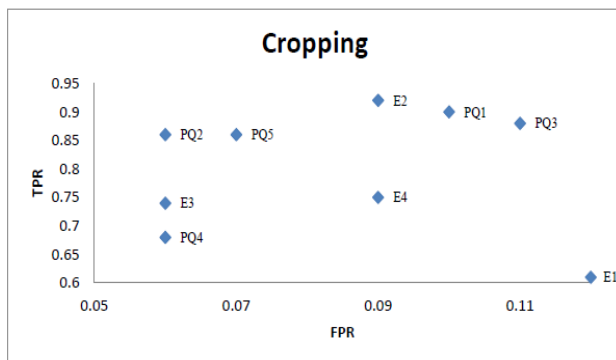


Fig. 4. ROC for DWT-CSLBP.

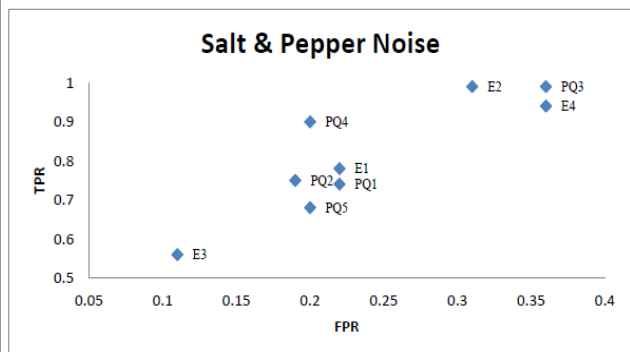
Observations: DWT-CSLBP have success rate of 87%. It shows only poor results for JPEG and average results for rotation attack. For all remaining attacks, it shows desirable discrimination power.

#### 4.2. Comparison of Proposed DWT-CSLBP Image Hashing and Existing Hashing Methods

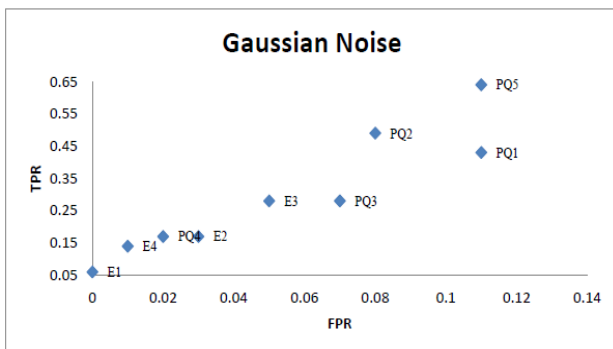
For comparative analysis, existing methods like LBP (256 bin), CSLBP (16 bin), CSLBP (Sep. Mag.) (64 bin), Q-CSLBP (8 bin) and our previous proposed 8 bin methods AQ-CSLBP, SDQ-CSLBP, CoCQ-CSLBP, LoGQ-CSLBP are taken into consideration. These existing methods and the proposed method are represented in Table 3 with their abbreviations. Results are displayed in terms of ROC for various attacks in Fig. 5.



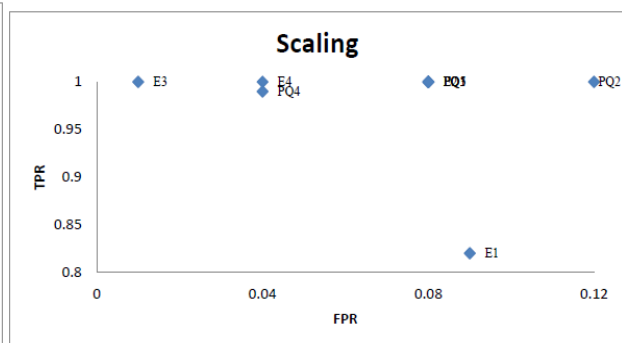
a) Cropping attack



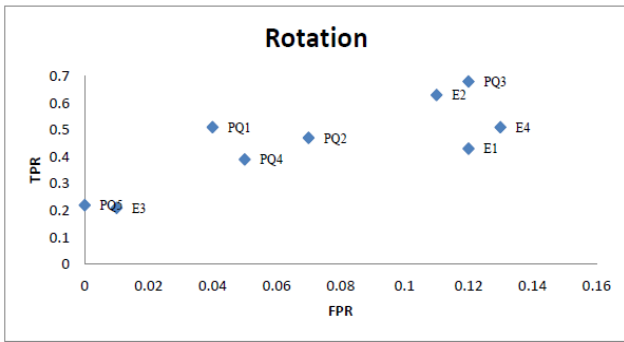
Salt & pepper noise attack



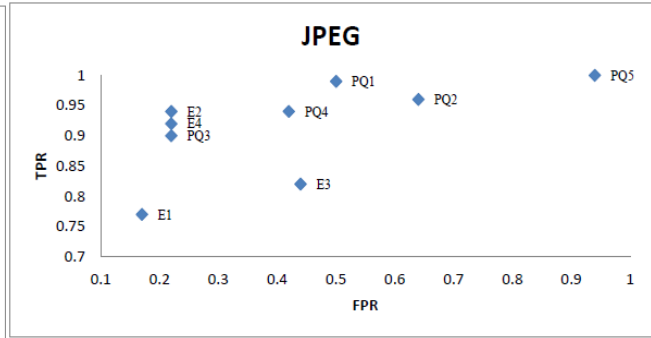
c) Gaussian attack



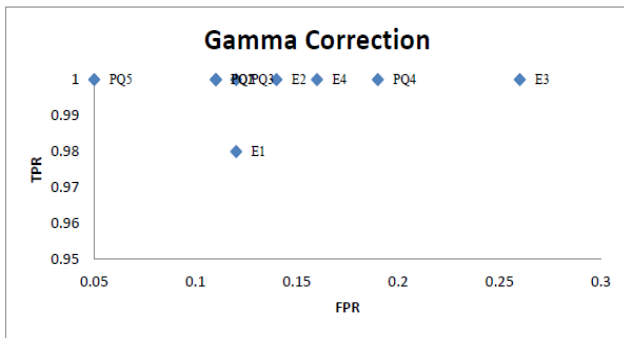
d) Scaling attack



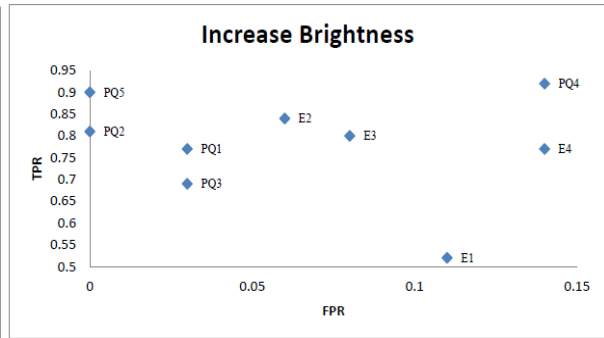
Rotation attack



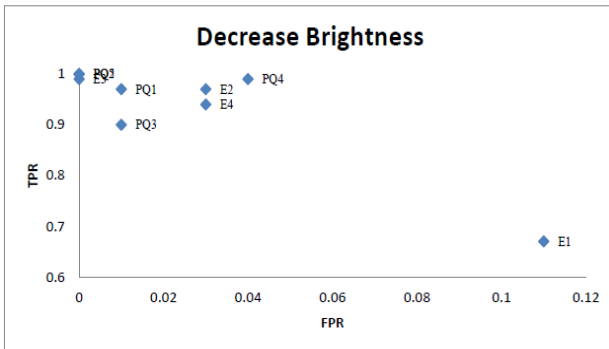
f) JPEG attack



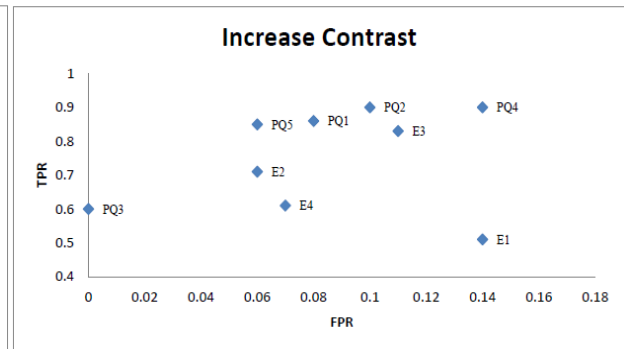
g) Gamma correction attack



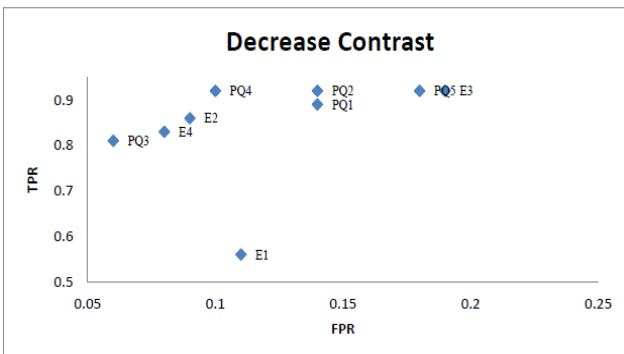
h) Increase brightness attack



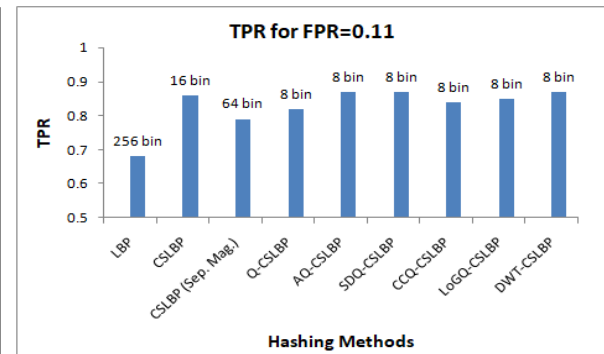
i) Decrease brightness attack



j) Increase contrast attack



k) Decrease contrast attack



l) TPR of proposed method and existing methods

Fig. 5. ROC results of the proposed method and comparative methods for various attacks.

## 5. Conclusion

We have proposed image hashing scheme based on combination of CSLBP and DWT. To increase discrimination quality of LL-CSLBP based image hash, boosting factor MASD is used during LL-CSLBP histogram construction. MASD represents Magnitude Average (MA) and Standard Deviation (SD) of local region of an image. Compact image hash is generated from LL band of DWT and desirable discrimination is obtained by weighted histogram. To store extracted features in simplified form, double bit quantization (DBQ) is used with minimum loss. Proposed method produces robustness to almost all types attacks despite compact length of image hash. However rotation results have little success rate. As a future work, we will consider extraction of rotation invariant feature and to incorporate in this method with minimum length.

## References

- [1] Mall, V., Bhatt, K., Mitra, S. K., & Roy, A. K. (2012). Exposing structural tampering in digital images. *Proceedings of International Conference on IEEE Signal Processing, Computing and Control (ISPCC)* (pp. 1-6).
- [2] Mall, V., Roy, A. K., Mitra, S. K., & Shukla, S. (2013). Detection of structural tampering in a digital image using canny edge detector. *Proceedings of International Conference on IEEE Informatics, Electronics & Vision (ICIEV)* (pp. 1-7).
- [3] Lamberger, M., Pramstaller, N., Rechberger, C., & Rijmen, V. (2008). Analysis of the hash function design strategy called SMASH. *IEEE Transactions on Information Theory*, 54(8), 1289-1306.
- [4] Swaminathan, A., Wu, M., & Liu, K. (2008). Digital image forensics via intrinsic fingerprints. *IEEE Trans. on Information Forensics and Security*, 3(1), 101-117.
- [5] Karsh, R. K., Laskar, R. H., & Richhariya, B. B. (2016). Robust image hashing using ring partition-PGNMF and local features. *Springer Plus*, 5(1), 1995.
- [6] Zhao, Y., Wang, S., Zhang, X., & Yao, H. (2013). Robust hashing for image authentication using zernike moments and local features. *IEEE Transactions on Information Forensics and Security*, 8(1), 55-63.
- [7] Soman, G., & John, J. K. (2016). Block-based forgery detection using global and local features. *Proceedings of International Conference on Springer Soft Computing Systems* (pp. 147-155).
- [8] Sebastian, L. S., Varghese, A., & Manesh, T. (2015). Image authentication by content preserving robust image hashing using local and global features. *Procedia Computer Science*, 46(1), 1554-1560.
- [9] Tang, Z., Huang, Z., Zhang, X., & Lao, H. (2017). Robust image hashing with multidimensional scaling. *Signal Processing*, 1375, 240-250.
- [10] Lei, Y., Wang, Y., Huang, J., Lei, Y., Wang, Y., & Huang, J. (2011). Robust image hash in radon transform domain for authentication. *Signal Processing: Image Communication*, 26(6), 280-288.
- [11] Swaminathan, A., Mao, Y., & Wu, M. (2006). Robust and secure image hashing. *IEEE Transactions on Information Forensics and security*, 1(2), 215-230.
- [12] Vadlamudi, L. N., Vaddella, R. P. V., & Devara, V. (2012). Robust image hashing technique for content authentication based on DWT. *Proceedings of International Conference on Springer Computer Vision and Image Processing* (pp. 1-6).
- [13] Prungsinchai, S., Kheli, F., & Bouridane, A. (2013). Fourier-Mellin transform for robust image hashing. *Proceedings of International Conference on IEEE Emerging Security Technologies (EST)* (pp. 58-61).
- [14] Plesca, C., & Morogan, L. (2014). Efficient and robust perceptual hashing using log-polar image representation. *Proceedings of International Conference on IEEE communications (COMM)* (pp. 1-6).
- [15] Monga, V. (2007). Robust and secure image hashing. *IEEE Transactions on Information Forensics and security*, 2(3), 376-390.

- [16] Tang, Z., Wang, S., Zhang, X., Wei, W., & Su, S. (2008). Robust image hashing for tamper detection using non-negative matrix factorization. *IEEE Transactions on Information Forensics and Security*, 2(1), 18-26.
- [17] Tang, Z., Zhang, X., & Zhang, S. (2008). Robust perceptual image hashing based on ring partition and NMF. *IEEE Transactions on Knowledge and Data Engineering*, 26(3), 711-724.
- [18] Mo, Z. W., Zhu, Y. S., & Liu, Z. (2014). A hash-based image content authentication scheme for tamper detection using compressive sensing. *Proceedings of International Conference on IEEE Wavelet Active Media Technology and Information Processing (ICCWAMTIP)* (pp. 244-248).
- [19] Mall, V., Roy, A. K., & Mitra, S. K. (2013). Digital image tampering detection and localization using singular value decomposition technique. *Proceedings of International Conference on IEEE Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)* (pp. 1-4).
- [20] Tang, Z., Dai, Y., & Zhang, X. (2012). Perceptual hashing for color images using invariant moments. *Appl. Math*, 6(25), pp. 643S-650S.
- [21] Neelima, A., & Singh, K. M. (2015). A robust image hash function based on color and texture features of the image. *Proceedings of International Conference on IEEE Advanced Computing and Communication (ISACC)* (pp. 238-243).
- [22] Chen, Y., Yu, W., & Feng, J. (2014). Robust image hashing using invariants of Tchebichef moments. *Optik-International Journal for Light and Electron Optics*, 125(19), 5582-5587.
- [23] Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(7), 971-987.
- [24] Xiao, J., & Wu, G. (2011). A robust and compact descriptor based on center-symmetric LBP. *Proceedings of International Conference on IEEE Image and Graphics* (pp. 388-393).
- [25] Davarzani, R., Mozaari, S., & Yaghmaie, K. (2015). Image authentication using LBP-based perceptual image hashing. *Journal of AI and Data Mining*, 3(1), 21-30.
- [26] Baber, J., Satoh, S. I., Afzulpurkar, N., & Bakhtyar, M. (2012). Q-CSLBP: Compression of CSLBP descriptor. *Proceedings of International Pacific-Rim Conference on Springer Multimedia Information Processing* (pp. 513-521).
- [27] Patil, V., & Sarode, T. (2016). Image hashing using AQ-CSLBP with double bit quantization. *Proceedings of International Conference on IEEE Optoelectronics and Image Processing (ICOIP)* (pp. 30-34).
- [28] Patil, V., & Sarode, T. (2016). Image hashing by SDQ-CSLBP. *Proceedings of International Conference on IEEE Advances in Computing, Communications and Informatics (ICACCI)* (pp. 2057-2063).
- [29] Patil, V., & Sarode, T. (2016). Image hashing by CCQ-CSLBP. *Proceedings of International Conference on IEEE Advances in Computing, Communications and Informatics (ICACCI)* (pp. 73-78).
- [30] Patil, V., & Sarode, T. (2016). Image hashing by LoG-QCSLBP. *Proceedings of International Conference on ACM Communication and Information Processing* (pp. 124-1280).



**Varsha Patil** has received M.E. (computer engineering) degree from Mumbai University in 2007, pursuing Ph.D from Mumbai University, India. She has more than 12 years of experience in teaching. Currently working as an assistant professor in Computer Engineering Department at South Indian Graduate School of Technology, Mumbai. She is a life member (ISTE). Her areas of interest are image processing, signal processing and data mining, machine learning.



**Tanuja K. Sarode** has received M.E. (computer engineering) degree from Mumbai University in 2004, Ph.D from Mukesh Patel School of Technology, Management and Engineering SVKMs NMIMS University, Vile-Parle (W), Mumbai, India in 2010. She has more than 17 years of experience in teaching. Currently working as a professor and head in Dept. of Computer Engineering at Thadomal Shahani Engineering College, Mumbai. She is a life member (ISTE) and (IETE). Her areas of interest are image processing, signal processing and computer graphics. She has 170 papers in international conferences/journal to her credit.