

A New Robust Reference Image Hashing System

In previous chapter, a chaotic perceptual hashing technique has been discussed. Even the technique offers good robustness and authentication ability but there are some limitations regarding discrimination capability. This was expected because this method is not equipped with robust feature selection. Therefore, the main objective of the proposed work is to develop a new hashing technique based on global and local features to overcome the aforementioned limitation. It is pertinent that the robust perceptual feature detection is an important step in the hash generation. Therefore, KAZE features is employed for invariant feature extraction in the proposed hashing system. The most stable key points are obtained considering scale information. In addition, the statistical features are also obtained based on the log-polar mapped (LPM) reference images in wavelet domain. The combination of DWT and LPM essentially enhance the robustness against geometric attacks as well as different general image processing operations. The core idea is to normalize the host image first through a geometric normalization process. Image normalization is a technique to transform an image into a standard form such that this normalized form is independent of any possible geometric or non-geometric distortions applied on the image. Therefore, image normalization is considered to be a pre-processing step, which essentially reduces the effect of intentional/un-intentional distortions on the hash generation. The used normalization is achieved by computing the invariant geometric and central moments. The reference images are then obtained by the log-polar mapping in the wavelet domain. The reference image is formed using the directive contrast of the log-polar mapped wavelet coefficients. A combination of these two domains results into the LPM- DWT, which exhibits the multiresolution property, describing the spatial as well as the transform domain information. The significant information of the reference image is obtained using the singular value decomposition to form an intermediate hash value. A randomization process is then proposed and performed on the intermediate hash value to construct the final hash sequence. The proposed hashing approach improves the robustness by reducing the effect of noise and geometrical distortions. This fact is further demonstrated by the experimental results, which in essence validates the excellent robustness and security of the proposed hashing technique against a variety of intentional/un-intentional distortions.

4.1 LOCAL IMAGE CONTRAST

The HVS perception states that it is the contrast between the objects rather than the intensities which differentiates the bright and dark image regions. Contrast essentially defined as the difference between intensities of the target and background, and it is usually defined as follows [Bhatnagar *et al.*, 2015; Whittle, 1986] :

$$C = \frac{\vartheta_t - \vartheta_b}{\vartheta_b} \quad (4.1)$$

where ϑ_t and ϑ_b are the intensity of target and background respectively. The aforementioned definition of contrast needs to be applied to a local area rather than the whole image. Thus, it is also said the local contrast of the image. In general, ϑ_b is referred to the local low frequency component while $\vartheta_t - \vartheta_b = \vartheta_h$ considered as the local high frequency.

4.2 PROPOSED HASHING SYSTEM

The proposed image hashing system consists of four procedures, image preprocessing, reference image, perceptual feature extraction and hash generation process. The first procedure obtains an invariant image to the different image manipulation. The second and third procedures essentially generate a reference image and extract the perceptual which is finally used in the fourth procedure to generate the hash sequence.

4.2.1 Reference Image Generation

Let I be a gray-scale image of size $m \times n$, whose reference image has to be generated. The complete process of reference image generation can be summarized as follows.

1. Perform ℓ -level wavelet transform (DWT) on I denoted by \mathcal{J}_l^θ where $l \in [1, \ell]$ is the level and $\theta \in \{A, H, V, D\}$ is the orientation of the sub-bands.
2. Select the approximate sub-bands \mathcal{J}_ℓ^A at the finest level ℓ .
3. Perform 1-level wavelet on \mathcal{J}_ℓ^A , which is denoted by $\mathcal{J}_{\ell+1}^\theta$ such that $\theta \in \{A, H, V, D\}$.
4. Find the local contrast, as defined in Eqn. 4.1, of all the high-frequency sub-bands as follows.

$$\text{Horizontal Contrast} : C^H = \frac{\mathcal{J}_{\ell+1}^H}{\mathcal{J}_{\ell+1}^A} \quad (4.2)$$

$$\text{Vertical Contrast} : C^V = \frac{\mathcal{J}_{\ell+1}^V}{\mathcal{J}_{\ell+1}^A} \quad (4.3)$$

$$\text{Diagonal Contrast} : C^D = \frac{\mathcal{J}_{\ell+1}^D}{\mathcal{J}_{\ell+1}^A} \quad (4.4)$$

5. Initialize high-frequency components to zero whose local contrast are less than a pre-defined threshold. The threshold is given by

$$T^\zeta = \text{Sort}(p * S^\zeta) \quad (4.5)$$

where $\zeta \in \{H, V, D\}$, $\text{Sort}(\circ)$ is the sorted local contrast, S^ζ is the size of subband $\mathcal{J}_{\ell+1}^\zeta$ and p is the percentage of the wavelet coefficients which are to be retained.

6. Perform 1-level inverse wavelet transform to construct reference approximate sub-band $\mathcal{J}_\ell^{A,ref}$, after setting all high-frequency coefficients to zero as defined in the previous step.
7. Perform inverse ℓ -level inverse wavelet transform to get the reference image denoted by I^{ref} .

4.2.2 Perceptual Feature Extraction

The local features are selected based on KAZE features as described in Section 2.7. These features are invariant to rotation, scaling and translation of the image. The steps used in features extraction are briefly explained as follows.

1. For input image (I), the key points are detected based on KAZE features. Let $Q = \{F_p | p = 1 \cdots k\}$ represent the key points, where $F_i = (S_i(x_i, y_i), \sigma_i, \delta_i)$. Here (x_i, y_i) are the respective coordinate position of the S_i whereas σ_i and δ_i is the corresponding scale and orientation factor.
2. Select k most stable points based on sigma value. The larger value of sigma indicates the higher stability.

3. The key-points having same scale factor are removed.
4. Construct two set $\theta_S = \{S_1, S_2, \dots, S_k\}$ and $\theta_f = \{(x_i, y_i), \sigma_i | i = 1 \dots k\}$ from the selected key points.
5. Perform 1-level db1 wavelet transform on the first set θ_S and Obtain a new set θ_S^N considering the Steps 3-7 of Sub-section 4.2.1.
6. Obtain a vector $\theta_{S,f}$ by concatenating both the vectors θ_S^N and θ_f .
7. Take absolute value of $\theta_{S,f}$ and round it to the nearest integer value.
8. Obtain a vector $H_{S,f}$ using binary conversion followed by 8 bit representation.

4.2.3 Hash Sequence Construction

In this sub-section, the proposed hash sequence generation process is discussed in detail. Considering, an image I as the input image, the proposed hash sequence generation approach consists of the following steps:

- Ensure that the input image is a gray-scale image. If not, convert the input image I to the gray-scale image by the luminosity method. Mathematically, luminosity method is defined by the following equation .

$$\tilde{I} = 0.3R + 0.59G + 0.11B \quad (4.6)$$

- Apply the pre-processing on the image \tilde{I} as mentioned in the Sub-section 3.1.1 to generate a normalized image \tilde{I}_N .
- Transform the normalized image into Log-polar coordinate system as defined in Section 2.5 . Let the transformed image is denoted by \tilde{I}_N^p . Semantically,

$$\tilde{I}_N^p = LPT \{ \tilde{I}_N \} \quad (4.7)$$

- Construct a reference image ($\tilde{I}_{N,ref}^p$) from \tilde{I}_N^p using the process defined in Sub-section 4.2.1.
- Perform SVD on the reference image $\tilde{I}_{N,ref}^p$.

$$\tilde{I}_{N,ref}^p = U_{\tilde{I}_{N,ref}^p} S_{\tilde{I}_{N,ref}^p} \left(V_{\tilde{I}_{N,ref}^p} \right)^T \quad (4.8)$$

- Generate a feature matrix by stacking first k columns of $U_{\tilde{I}_{N,ref}^p}$ and $V_{\tilde{I}_{N,ref}^p}$.

$$R_f = \left[U_{R_f}^{(1)}, \dots, U_{R_f}^{(k)}, V_{R_f}^{(1)}, \dots, V_{R_f}^{(k)} \right] \quad (4.9)$$

- Again, perform SVD on the feature matrix R_f .

$$R_f = U_{R_f} S_{R_f} V_{R_f}^T \quad (4.10)$$

- Construct a feature vector F_R by stacking first columns of U_{R_f} and V_{R_f} as follows.

$$F_R = \left[U_{R_f}^{(1)}, V_{R_f}^{(1)} \right] \quad (4.11)$$

- Obtain the threshold T by taking the mean value of the feature vector F_R .
- Then obtain a binary sequence H^f as

$$H^f(i) = \begin{cases} 1, & F_R \geq T \\ 0, & \text{otherwise} \end{cases} \quad (4.12)$$

- Obtain the hash sequence by concatenating of the sequence H^f and $H_{S,f}$.
- Final hash value is obtained by the randomization of the hash sequence (H). Assuming the length of H is \hbar , the randomization process can be summarized as follows.

1. Stack the sequence (H) into an array (A_H) of size $p \times p$, where $p = \text{sqrt}(\hbar)$.
2. Shuffle A_H using Arnold cat map, which is denoted by A_H^S . Mathematically, the classical cat-map is defined as follows [Arnol'd and Avez, 1968; Sui and Gao, 2013]

$$\begin{bmatrix} x'_n \\ y'_n \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \pmod{\ell} \quad (4.13)$$

where the pair (x'_n, y'_n) is the coordinates in the shuffled array A_H^S corresponding to the coordinates (x_n, y_n) of the original array A_H .

3. Arrange the A_H^S in to a vector to form the final hash sequence H_I .

4.3 EXPERIMENTAL RESULTS AND DISCUSSION

It is planned to evaluate the proposed image hashing algorithm from four aspects. The first one is their perceptual robustness against content-preserving manipulations, it is important for content-based image identification and retrieval wherein it is desired that perceptually identical images under distortions would have similar hashes. Secondly, the robustness of image authentication, which is used to determine whether the received image has the same contents as the trusted one or has been maliciously tampered, or utterly peculiar. The third one is the unpredictability, a

Table 4.1: Normalized Hamming distance of different standard test images.

Images	Lena	Barbara	Cameraman	Mandrill	Watch	Pirate	Pepper	House	Hall	Car
Lena	0	0.390	0.375	0.5375	0.395	0.5325	0.6625	0.3425	0.4325	0.615
Barbara	0.390	0	0.4000	0.6175	0.45	0.7125	0.4675	0.4475	0.4725	0.575
Cameraman	0.375	0.4	0	0.4325	0.305	0.6075	0.4675	0.3525	0.5025	0.7
Mandrill	0.5375	0.6175	0.4325	0	0.3525	0.43	0.305	0.4	0.435	0.3925
Watch	0.395	0.45	0.305	0.3525	0	0.4625	0.4675	0.3375	0.3125	0.47
Pirate	0.5325	0.7125	0.6075	0.43	0.4625	0	0.425	0.545	0.43	0.3275
Pepper	0.6625	0.4675	0.4675	0.305	0.4675	0.425	0	0.39	0.505	0.5225
House	0.3425	0.4475	0.3525	0.4	0.3375	0.545	0.39	0	0.48	0.7325
Hall	0.4325	0.4725	0.5025	0.435	0.3125	0.43	0.505	0.48	0	0.3925
Car	0.615	0.575	0.7	0.3925	0.47	0.3275	0.5225	0.7325	0.3925	0

necessary property of a secure hashing algorithm. Conclusively, the final aspect is the computation cost.

The efficiency of the proposed scheme has been tested using the standard test images including Barbara, Cameraman, Car, Hall, House, Lena, Mandrill, Pepper, Pirate and Watch for quantitative assessment. Normalized hamming distance [NHD] is used to measure the similarity between the image hash. For a pair of hash sequence, the normalized hamming distance can be defined as:

$$d(h_1, h_2) = \frac{1}{N} \sum_{j=1}^N \delta(h_1(j), h_2(j)) \quad (4.14)$$

where

$$\delta(h_1(j), h_2(j)) = \begin{cases} 1, & h_1(j) = h_2(j) \\ 0, & h_1(j) \neq h_2(j) \end{cases} \quad (4.15)$$

where N is the length of the hash sequence, h_1 and h_2 are hash sequences with their corresponding j^{th} elements $h_1(j)$ and $h_2(j)$. The estimated NHD is then used to characterize similar and dissimilar images. In principle, if this value is less than some predefined threshold then image is considered to be perceptually similar otherwise perceptually different images. The NHD for the standard images have been shown in Table 4.1.

4.4 ROBUSTNESS TEST

The robustness of the proposed hashing scheme is examined through a variety of attacks such as additive Gaussian noise, Salt & pepper noise, blurring, average and median filtering, scaling, rotation, shearing, brightness correction and contrast adjustment, JPEG and SPIHT compression. To assess the performance, a database of 1000 images have been constructed comprising of 100 benchmark images (such as Barbara, Lena, Cameraman, etc.) of USC-SIPI database and a variety of natural images (such as building, landscape, animal and fruits), where each image is converted to gray-scale image and down-sampled to 256×256 . Furthermore each image is passed to a variety of standard content preserving operations as listed in the Table 4.2. Then, normalized hamming distance is evaluated between the hash of original and manipulated images. The averaged performance of the proposed scheme are compared with some of classical schemes namely, Tang method [Tang *et al.*, 2019], Ouyang methods [Ouyang *et al.*, 2016, 2017], Wang method [Wang *et al.*, 2015] and Qin method [Qin *et al.*, 2013]. In the proposed scheme, the hash value is based on local and global features, therefore the performance is compared with both type of schemes consisting global and local & global features. For fair comparison, the results are evaluated using normalized hamming distance. The usual results can be seen in Fig. 4.1. It can be observed that proposed scheme estimates better performance over the existing schemes specifically geometric distortions. The effectiveness of proposed scheme is also analyzed for noisy and filtering operations and are shown in Fig. 4.1(a-e). The results indicate that the performance of proposed scheme is comparable with that of existing schemes. In addition, performance of the hashing scheme is also examined against contrast and brightness adjustment, JPEG and SPIHT compression. The respective results are shown in Fig. 4.1(i-l). For contrast adjustment, performance is better among all and comparable with that of other ones. In essence, the results indicate that the proposed hashing algorithm achieves good robustness for all types of manipulations.

To further validate the efficiency, the performance of the proposed scheme is explored using a set of 106 visual identical images which essentially generate the $1000 \times 106 = 106,000$ identical image pairs wherein the list of used operations are displayed in Table 4.2. Then normalized hamming distance is computed between the hash of original and visually identical images. The

Table 4.2 : Content-Preserving Operation with Different Parameter Details.

Types of Manipulation	Tools	Operations	Descriptions	Parameters Range	Number of Images
Noise Addition	MATLAB	Gaussian noise	Noise variance	[0.010, 0.030]	5
	MATLAB	Salt & Pepper noise	Noise density	[0.020, 0.100]	5
	MATLAB	Speckle noise	Noise density	[0.020, 0.100]	5
Filtering Operation	MATLAB	Median filter	Windows size	[3,5,7,9,11]	5
	MATLAB	Average filter	Windows size	[3,5,7,9,11]	5
	MATLAB	Gaussian filter	Windows size	[3,5,7,9,11]	5
Geometric Operation	MATLAB	Cropping	Area (in percentage)	[1, 5]	5
	MATLAB	Rotation	Rotation angle	[5, 25]	5
	MATLAB	Scaling	Scaling factor	[0.5, 1.5]	10
Image Processing Operation	MATLAB	Shearing	Ratio	[0.1, 0.1]	5
	MATLAB	Row-deletion	Number of row	[2, 10]	5
	MATLAB	Un-Zign	Number of row	[2, 10]	5
Lossy Operation	MATLAB	Contrast adjustment	Ratio	[.9,1.1]	10
	MATLAB	Brightness adjustment	Ratio	[.9,1.1]	10
	MATLAB	Gamma correction	Gamma value	[0.8, 1.3]	5
Total	MATLAB	JPEG Compression	Quality factor	[10, 80]	8
	MATLAB	SPIHT Compression	Quality factor	[10, 80]	8
Total					106

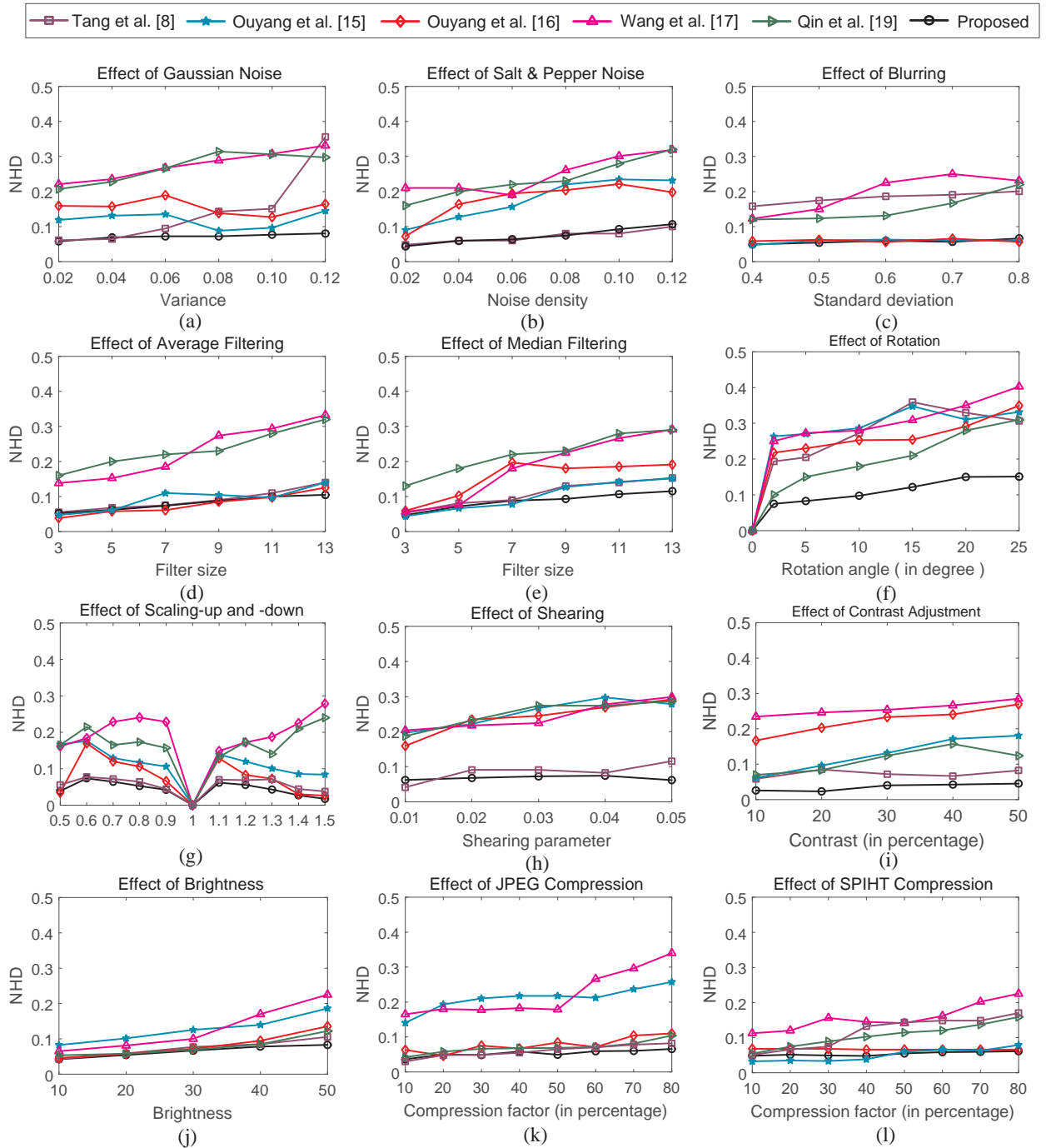


Figure 4.1: Robustness and comparative analysis of the proposed hashing technique under different distortions: (a) Additive Gaussian noise, (b) Salt & pepper noise, (c) Image blurring, (d) Average Filtering, (e) Median Filtering (f) Rotation, (g) Scaling, (h) Shearing, (i) Contrast Adjustment, (j) Brightness, (k) JPEG Compression, (l) SPIHT Compression.

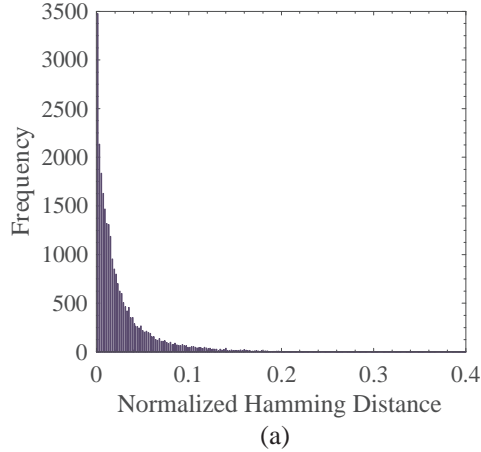


Figure 4.2 : Distribution of normalized hamming distance for visually identical image pairs.

distribution of NHD obtained from perceptually identical and distinct image pairs are depicted in Fig. 4.2, where the frequency (y -axis) of a NHD (x -axis) is plotted. From the figure, it can be observed that the minimum and maximum NHD are approximately (0,0.15) and (0.2, 0.8) respectively for perceptually identical and distinct image pairs. The NHD value close to 0.15 indicates that 0.02% distinct images falsely recognized as identical images. This essentially includes the images obtained from closely and severely effected content preserving manipulations. In contrast, NHD value close to 0.20 indicates that 0.01% identical images are recognized as distinct images. Therefore, the proposed scheme works well against content preserving manipulations proving the robustness of the proposed hashing scheme.

4.4.1 Discriminative capability

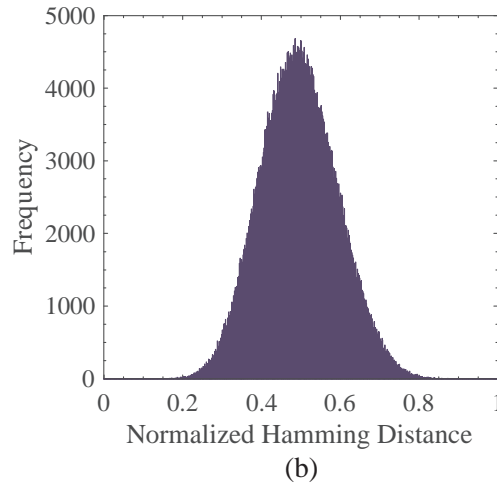
Discriminative or anti-collision property implies the capability of generating the similar hashes with very low probability from perceptually distinct images. To analyze the anti-collision capability, the normalized hamming distance between the different pairs of standard images are calculated, which are shown in Table 4.1. It can be observed from the table that the maximum NHD is 0.6328 and the corresponding pair of images is (House, Pirate) whereas the minimum hamming distance is 0.3008 and the corresponding pair of images are (Barbara, Watch), which shows that proposed hashing scheme has good anti-collision capability. This process is further extended to 1000 different images carrying distinctive contents, such as human beings, scenery, buildings, monuments and sport, and their sizes are between 512×512 to 900×800 . This collection will have 4,99,500 pairs in total and the NHD is computed for each pair. The probability distribution of NHD is obtained exploiting chi-square test, which comes out to be normal distribution, which is depicted in Fig. 4.3. The mean and standard deviation of the distribution are 4.8 and 0.2 respectively. The probability of a hash distance less than a threshold (λ) defines the collision probability CP_r , and can be determined as follows:

$$\begin{aligned}
 CP_r(NHD \leq \lambda) &= \frac{1}{\sqrt{2\pi}\sigma} \int_0^\lambda \exp\left[-\frac{(z-\mu)^2}{2\sigma^2}\right] dx \\
 &= \frac{1}{2} \operatorname{erfc}\left(-\frac{\lambda-\mu}{\sqrt{2}\sigma}\right)
 \end{aligned} \tag{4.16}$$

where $\operatorname{erfc}(\cdot)$ represent the error function. The collision probability are computed at different thresholds and are shown in Table 4.3. Clearly, if the threshold is decreasing, then the collision

Table 4.3 : Collision Probabilities for Different Thresholds (λ)

Threshold (λ)	Collision probability
0.30	8.2967×10^{-3}
0.25	5.2717×10^{-3}
0.20	1.2988×10^{-3}
0.15	1.6783×10^{-4}
0.10	2.7972×10^{-5}
0.05	3.9960×10^{-6}

**Figure 4.3 :** Distribution of normalized hamming distance for perceptually different image pairs.

probability is also decreasing, which means a low probability of false classification of distinct images to similar images, i.e., a good discrimination. However, the robustness and performance of the hashing system may be compromised in case of smaller threshold values in the sense that the rate of falsification may increase for content preserving operations. Therefore, an optimal threshold may be chosen to make a trade-off between discrimination and robustness.

4.4.2 Unpredictability of image hash

This section presents the security analysis of the proposed hashing scheme in the sense of ability to predict the hash. In essence, it indicates the unpredictability of the hash under brute force attacks. For this purpose, 2000 binary hash sequences are randomly generated with equal probabilities of symbols 0's and 1's, i.e., $P(1) = P(0)$ wherein the length of these hash sequence is same as the original image hash. The normalized hamming distance is then determined between the true hash values and the generated random sequences. The NHD curves are displayed in Fig. 4.4. It can be observed that all the hamming distances are close to 0.5. This essentially reveals the unpredictability of the image hash values. Therefore, it can be concluded that the proposed hashing has the capability of resisting against brute force attacks.

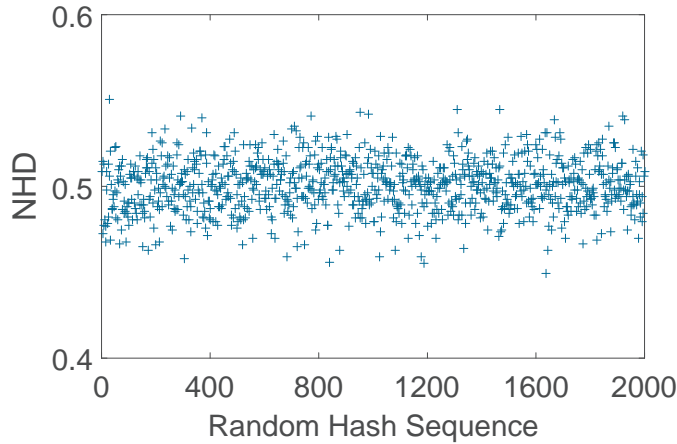


Figure 4.4 : The normalized hamming distance between true hash sequence and randomly generated binary sequence

4.5 SENSITIVITY ANALYSIS

The images are widely used nowadays in various fields like forensics, computerized photography, astronomy, medical imaging, character recognition and remote sensing. These images are frequently altered/tempered to cater the specific needs of the intended application. Therefore, the use of malicious tempering may lead to serious impact on the many aspects of societal security including authentication and integrity of digital data. This is due to the fact that the maliciously tempering using the insertion, deletion and/or copy-paste operations may lead to perceptually different images. Therefore, an ideal perceptual hash function must be sensitive to the content preserving malicious modification and has the capability to recognize the malicious alterations. In principle, the hashes corresponding to tempered image must be different than that of the original image. To validate the sensitivity of the proposed system, some tempering operations are employed on the images. For this purpose, the most common tampering operations like “deletion”, “insertion”, “copy and move”, “copy and paste” and “manual tempering” of the object are carried out on different images. Each of these tempering has been done from two different point of views (1) tempering of similar objects, and (2) tempering of different objects present in the image. The images considered for the sensitivity analysis and corresponding forged versions are depicted in Fig. 4.5. In the first row, three different objects of “football match” image is deleted one by one wherein it can be noticed that even the minute manipulation of the image results into in-authenticated image. A similar observation can be drawn from the images depicted in the remaining rows of the Fig. 4.5. This indicates that the proposed hashing system is highly sensitive to small alteration as well as larger manipulation.

4.5.1 Computation Complexity

Computational complexity is another metric which is used to gauge the performance of a hashing system. It refers to the time required in the estimation of the hash value of the multimedia data. The average execution time is measured for the hash generation over 1000 images used in the discrimination test. For comparative analysis, proposed and existing methods [Ouyang *et al.*, 2016, 2017; Qin *et al.*, 2013; Tang *et al.*, 2019; Wang *et al.*, 2015] are each algorithm is run on a PC configured with Windows-7, Core i5 processor under MATLAB platform. The respective average computational time for the proposed and existing schemes are listed in Table 4.4. It can be seen from the table that the proposed scheme takes a little longer time than [Ouyang *et al.*, 2016] and lesser time than that of methods presented in [Ouyang *et al.*, 2017; Qin *et al.*, 2013; Tang *et al.*, 2019;


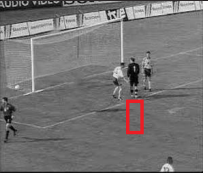
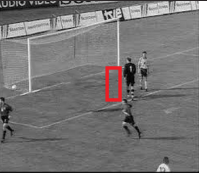
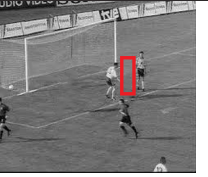

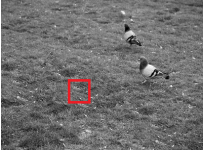
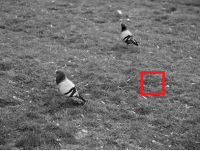
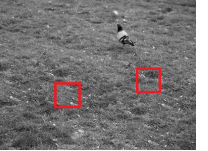
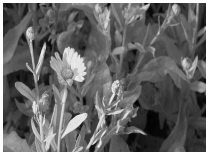






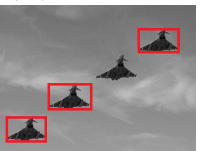
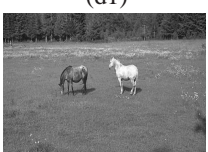



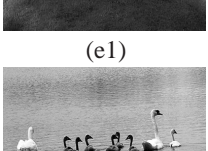
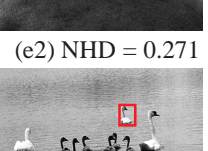
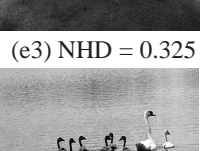
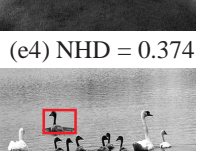
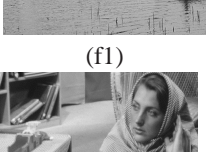
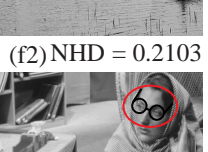

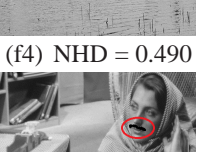

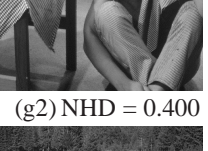
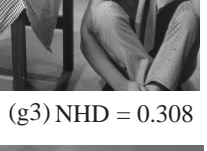
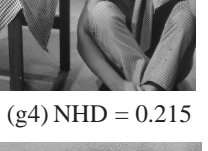
Operation	Original Image	Forged Image1	Forged Image2	Forged Image3	
Deletion of object from the test image	 (a1)	 (a2) NHD = 0.300	 (a3) NHD = 0.214	 (a4) NHD = 0.280	
	 (b1)	 (b2) NHD = 0.268	 (b3) NHD = 0.212	 (b4) NHD = 0.342	
	 (c1)	 (c2) NHD = 0.488	 (c3) NHD = 0.220	 (c4) NHD = 0.454	
	 (d1)	 (d2) NHD = 0.211	 (d3) NHD = 0.251	 (d4) NHD = 0.474	
Copy and move attack	 (e1)	 (e2) NHD = 0.271	 (e3) NHD = 0.325	 (e4) NHD = 0.374	
	 (f1)	 (f2) NHD = 0.2103	 (f3) NHD = 0.202	 (f4) NHD = 0.490	
	 (g1)	 (g2) NHD = 0.400	 (g3) NHD = 0.308	 (g4) NHD = 0.215	
	 (h1)	 (h2)	 (h3)	 (h4)	
Object replacement		NHD(g1,h1) = 0.2857	NHD(e1,h2) = 0.4429	NHD(d1,h3) = 0.5486	NHD(f1,h4) = 0.2314

Figure 4.5 : Original image and corresponding maliciously modified version

Wang *et al.*, 2015]. This is mainly due to the fact that method in [Ouyang *et al.*, 2016] utilizes moment based features whereas proposed method computes both local and global features for final hash estimation.

Table 4.4 : Time complexity of different hashing algorithms.

Hashing Scheme	Average Time
Tang Method	3.1189
Ouyang Method	1.702
Ouyang Method	2.17
Wang Method	1.981
Qin Method	3.47
Proposed Method	1.8932

Table 4.5 : Estimated performance of different hashing algorithms.

	Tang Method	Ouyang Method	Ouyang Method	Wang Method	Qin Method	Proposed Method
Features	Global	Local and Global	Global	Local and Global	Global	Local and Global
Robustness against noise addition	Yes	Yes	Yes	Yes	No	Yes
Robustness against angle rotation	No	No	No	No	No	Yes
Robustness against cropping	No	Yes	No	Yes	No	Yes
Robustness against Compression	Yes	Yes	Yes	Yes	Yes	Yes
Content Sensitivity	Average	Average	Poor	Good	Poor	Good

4.6 PERFORMANCE COMPARISON

The efficiency of the proposed hashing scheme is evaluated by comparing it with different state-of-the-art methods in the sense of tempering. In this connection, the experimental image is tempered in following ways: (1) Deletion of objects from the image, (2) Insertion of objects in the image, (3) copy and move objects in the image, (4) copy and paste objects in the image, (5) manual tempering, and (6) object replacement in the image. The visual assessment of these tempering can be seen in Fig. 6, wherein it can be observed that the proposed hashing scheme has the ability to differentiate between the perceptually similar and perceptually different images. Further, the same observation can be drawn for the content preserving operations (CPOs) or content changing operations (CCOs). A qualitative comparison is depicted in Table 4.5, which essentially indicates that proposed system shows excellent robustness against different kind of content preserving manipulations in comparison to existing technique. Among all hashing schemes, schemes presented in [Ouyang *et al.*, 2017; Qin *et al.*, 2013; Tang *et al.*, 2019] disseminate the baseline results providing average performance for almost all the experimental images. This is anticipated since these schemes are based on the global features only. This limitation is rectified in [Ouyang *et al.*, 2016; Wang *et al.*, 2015], where authors tried to combine global and local features, however, these schemes suffer from the bad synchronization between global and local features. On the contrary, the performance of proposed scheme is better than that of existing hashing schemes. This is in light of the fact that proposed scheme uses KAZE features as local features, which essentially has distinction of using non-linear scale-space over the SIFT features (which are used in [Ouyang *et al.*, 2016; Wang *et al.*, 2015] as local features).

The effectiveness of the proposed scheme is further verified by insightful comparison with existing schemes using receiver operating characteristic (ROC) curve [Fawcett, 2006]. The ROC curve has ability to capture the trade-off between robustness and discrimination. Therefore, ROC

Table 4.6 : Comparative analysis between proposed and existing algorithms.

Performance	Tang Method	Ouyang Method	Ouyang Method	Wang Method	Qin Method	Proposed Method
TPR when FPR ≈ 0	0.8916	0.972	0.9166	0.9340	0.6812	0.9980
FPR when TPR ≈ 0	0.3833	0.208	0.4613	0.0620	0.5221	0.0044
AUC	0.9612	0.9843	0.9655	0.9951	0.9414	0.9998
Hash	96	384	96	340	444	350

curve is widely used for perceptive comparative study of different hashing frameworks. The ROC curve essentially estimates the relationship between the true positive rate (TPR) and false positive rate (FPR). Mathematically, it can be determined as follows:

$$TPR(\lambda) = \frac{q_1(NHD < (\lambda))}{Q_1}, FPR(\lambda) = \frac{q_2(NHD > (\lambda))}{Q_2} \quad (4.17)$$

For a predefined threshold (λ), q_1 represents the number of identical image pairs categorized as similar images whereas q_2 is the number of different or malicious modified image pairs categorized as similar ones. On the other hand, Q_1 and Q_2 respectively are the total number of visually identical and different images.

The ROC curve of proposed and existing techniques have been plotted for all manipulations and depicted in Fig. 4.6. From the figure, it can be observed that ROC curve higher among all techniques and converge towards the top left corner. This reveals that the proposed technique is most effective and efficient among all state-of-the-art schemes considered in the comparative analysis. Additionally, area under curve (AUC) is also calculated to verify the same. A quantitative comparison is summarized in Table 4.6. From the Table, it can be observed that proposed scheme provides better classification in comparison to existing schemes.

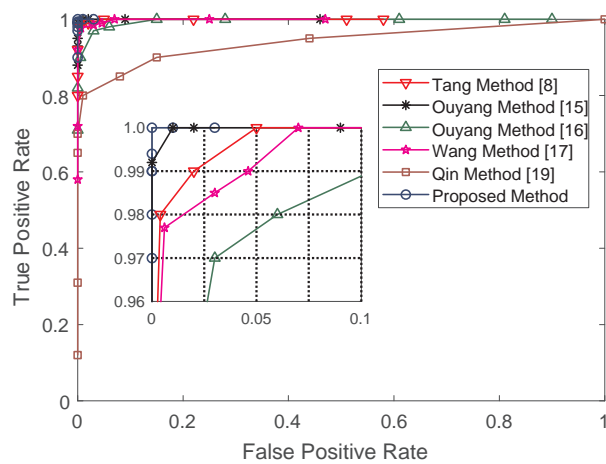


Figure 4.6 : Comparison of ROC curve of different existing techniques

Finally, performance of the proposed technique is also compared with the existing hashing techniques in terms of the content sensitivity. For this purpose, the perceptual similarity is evaluated for the images shown in Fig. 4.5. The optimal threshold corresponding to FPR ≈ 0 , have been

Table 4.7 : Content sensitivity analysis among different algorithm.

Algorithm	Threshold	Normalized Hamming distance between the original and tempered images							
		(a1,a2)	(b1,b2)	(c1,c2)	(d1,d2)	(e1,e2)	(f1,f2)	(g1,g2)	(f1,g1)
Tang Method	10	2.6861	3.6914	1.1189	3.4580	3.7392	4.5113	1.0532	5.1676
Ouyang Method	10	3.6056	9.5917	9.8489	8.2462	5.0990	11.4891	4.4721	7.9373
Ouyang Method	10	4.2426	2.4495	3.6056	4.3589	2.0000	4.5826	4.3613	1.7321
Wang Method	0.49	0.4922	0.4883	0.4824	0.2988	0.2988	0.3672	0.3613	0.4315
Qin Method	0.20	0.0566	0.0731	0.0367	0.1463	0.1102	0.1220	0.0452	0.1629
Proposed Method	0.20	0.3000	0.2685	0.4880	0.2110	0.2714	0.2100	0.4000	0.2857

selected for the best fit comparison. A detailed comparison can be seen in Table 4.7. Due to page limitation, the detailed comparison is only presented for first (original images) and second (respective tempered image 1) column of Fig. 4.5. It can be observed that the content sensitivity of method in [Qin *et al.*, 2013] is very poor among all considered techniques. In contrary, methods presented in Ouyang *et al.* [2017]; Tang *et al.* [2019] have low sensitivity against small content modifications, whereas methods in Ouyang *et al.* [2016]; Wang *et al.* [2015] have moderate level of sensitivity. In contrast, proposed technique not only preserve good sensitivity but also authenticate the images perfectly even when the image is severely modified. Therefore, it can be concluded that the proposed hashing scheme shows better classification performance when compared to state-of-the-art hashing schemes.

4.7 SUMMARY

In this work, an efficient image hashing scheme has been developed for image content authentication based on perceptual and statistical features. The perceptual information are obtained based on KAZE features whereas a reference image and log-polar mapping are utilized for estimating the statistical features. The proposed hashing scheme is robust to content preserving (such as noise addition, filtering, gamma correction, contrast and sharpness adjustment) and geometric (such as rotation, scaling and SPIHT compression) operations. Extensive experimental and comparative analysis have been conducted to confirm the superiority and efficiency of the proposed hashing scheme in terms of anti-collision capabilities, content sensitivity, discriminative capabilities, unpredictably, sensitivity and robustness against a variety of malicious modifications.