Received 29 October 2023, accepted 4 November 2023, date of publication 8 November 2023,
date of current version 16 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3330969

RESEARCH ARTICLE

Content Authentication and Tampered Localization Using Ring Partition and CSLBP-Based Image Hashing

ABDUL S. SHAIK[1,2], RAM K. KARSH[1], (Senior Member, IEEE), MOHIUL ISLAM[3], AND SHUBHASHISH BHAKTA[4]

[1]Department of Electronics and Communication Engineering, National Institute of Technology (NIT) Silchar, Silchar, Assam 788010, India
[2]Department of Electronics and Communication Engineering, CMR College of Engineering and Technology, Hyderabad, Telangana 501401, India
[3]School of Electronics Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu 632014, India
[4]School of Electrical Engineering and Computing, Electrical Power and Control Engineering (EPCE) Program, Adama Science and Technology University, Adama, Ethiopia

Corresponding author: Shubhashish Bhakta (bhaktamelt@gmail.com)

ABSTRACT In the present world, innovation, and creative thinking were pivotal in bringing about the digital revolution. As a result of consistent development in multimedia processing algorithms, altering the contents of digital images has become more accessible. Image hashing has been found to be one of the most suitable approaches for content authentication applications. A ring partition and center-symmetric local binary patterns (CSLBP) approach has been introduced in the work for generating image hash. Our ring-based statistical characteristics remain unchanged when an image is rotated at any angle. As a result, it provides rotation invariant property. CS-LBP considers pairs of symmetrically opposite pixels around the central pixel. This symmetric encoding captures texture information that is invariant to certain rotations. The tampered detection and localization are done by computing the hash correlation between the original and tampered images. The experimental findings demonstrate that the suggested image hashing method can provide desirable robustness and discrimination in all content preserving operations (CPO). In comparison to specific current schemes, the proposed method offers shorter hash length, improved classification performance, and reduced hash generation time. Geometric rectification used in the suggested method can identify tampering even when geometric adjustments occur instantly. The receiver operating characteristics (ROC) curve demonstrates that the suggested model exhibits superior performance compared to other cutting-edge techniques.

INDEX TERMS Image hashing, ring partition, geometric correction, center symmetric local binary patterns, tampered detection, tampered localization.

I. INTRODUCTION

With the advancement in multimedia technology, the manipulation of digital images has become a major issue. The development of powerful image-altering tools encouraged us to investigate complex authentication systems. Recognizing genuine images from forgeries and locating the manipulated area is a difficult problem for industry and academia. Image hashing is a technique used in computer vision and image processing to generate compact representations (hash codes) of images that can be used for various tasks, such as image similarity comparison, duplicate detection, reverse image search, and content authentication. The main idea behind image hashing is to convert the complex visual content of an image into a fixed-length string of bits (the hash code) while preserving certain properties that allow for efficient and accurate comparisons between images. The image hashing introduced key benefits such as Efficient Comparison, Reduced Storage and Bandwidth, Content Authentication, Reverse Image Search, Privacy Preservation, Scalability, and

The associate editor coordinating the review of this manuscript and approving it for publication was Mohamad Afendee Mohamed.

Noise Tolerance. Numerous applications, including image authentication, tamper detection, image retrieval, and others, have used image hashing techniques.

The image hash from the ring partition process involves the following steps:

Ring Partitioning: The first step is to divide the image into circular or ring-shaped regions. The number and size of these regions can be predefined or adjusted based on the specific application or requirements.

Region Representation: Each circular region is then represented in a meaningful way. This representation can be based on color, texture, gradients, or other relevant image features. The goal is to capture the distinctive characteristics of the content within each region.

Hash Generation: After obtaining the region representations, a hash function is applied to each representation to generate a compact and fixed-size hash value. The hash value is a fingerprint of the image content within that particular region.

Aggregation: Finally, the individual hash values from all the regions are combined or aggregated to form a single hash value for the entire image. Various methods can be used for this aggregation, such as bitwise operations or concatenation.

The main contributions are as follows.

- ➢ The literature review suggests that most existing image-hashing techniques suffer from geometric transformation. Hence, a geometric transformation invariant algorithm has been proposed to address the issue.
- ➢ Unlike most existing techniques, the proposed image hashing algorithm can detect and localize tiny tampering areas in images. Rotation invariance property is achieved by considering circular regions for ring partitioning.
- ➢ In perceptual image hashing, computational speed plays a vital role. Preferring Local Binary Pattern (LBP) proves advantageous in this context due to its minimal computational complexity. An exclusive signature has been extracted from the input image to facilitate perceptual image hashing, capturing its visual characteristics.
- ➢ To achieve a high degree of discrimination, the center symmetric local binary pattern has been integrated as a merging approach which converts the input image into a grayscale invariant representation.

This paper has been organized as follows. The related work has been discussed in Section II. Section III describes the proposed image authentication system. Experimental results analysis and comparative analysis have been mentioned in Section IV. The work has been concluded in Section V.

II. RELATED WORK

Numerous researches have been conducted in the field of image hashing during the past few decades. In 2008, Tang et al. [1] presented an image-hashing technique using non-negative matrix factorization (NMF). This approach was designed for tamper detection applications [2]. However, a significant limitation of this method is its sensitivity to adjustments in brightness and contrast. In 2011, Tang et al. [3] introduced a structural feature-based method by integrating Discrete Cosine Transform (DCT) and the existing NMF. This method utilized structural features for image hashing and proposed a similarity measure for tampering detection. Despite its potential for tamper detection, this technique is susceptible to image rotation.

Similarly, in 2012, a few perceptual image hashing algorithms [4], [5], [6] were proposed based on Zernike moments, local features, shape contexts, and invariant moments to authenticate color images. Still, these methods fail to provide adequate performance in geometric attacks and tamper localization. Likewise, in 2013 and 2014, Tang and co-researchers [7], [8] introduced a robust image hash for image authentication using ring-based entropies and ring partitions with NMF. This technique can withstand JPEG compression, geometric distortion, noise addition, blurring, and image enhancement. However, the main drawback of this method is that it did not consider the fact that two visually distinct images could potentially share similar features.

In 2015, an image hashing approach [9] was proposed utilizing ring partition, projected gradient NMF, and local features. The algorithm is based on robust image hashing using ring partition (PGNMF). This method offers a secure image hash, robustness against image rotation, and desirable discriminative capabilities. However, this technique did not address the issue associated with color images. Similarly, in 2016, a few other techniques, such as resilient image hashing techniques employing ring partition, invariant vector distance, CSLBP, and PGNMF with local features [10], [11], [12], have been proposed. These approaches utilize the ring partition and local binary patterns as the foundation for image hashing. While these methods effectively withstand JPEG compression, gamma correction, and blurring, they lack the necessary resilience against image rotation.

Later, in 2017 and 2018, few robust image hashing based on DWT-SVD via a geometric correction and spectral residual method [13], [14], [15] were proposed. Their algorithms employ geometric corrections to establish the image hashing process. Although these techniques can endure content-preserving image manipulations, they can only tolerate rotations of up to 5°.

Similarly, in 2019, Tang, Qin, et al. [16], [17] presented a secure and better image hashing technique utilizing the Weber local binary pattern with a combination of color angle interpretation. It employed a method involving the tensor decomposition and Weber's local binary pattern. The drawback associated with these methods is the absence of tamper localization.

In 2020, alignment-based hashing approaches that utilize adaptive local feature extraction to enhance robustness in detecting tampering [18], [19] have been proposed. These methods rely on adaptive local features to create the image hash. These techniques are effective in identifying tampering and remain resilient against digital alterations. However, it fails to provide adequate performance in detecting color alterations within an image. In the same year, a few more robust perceptual image hashing techniques employing a color structure, Fractal coding, and ordinal measures [20], [21], [22], [23] were proposed. These methods proposed a color structure with intensity gradient, Laplacian Pyramids, and ordinal measures for creating the hash. These approaches demonstrate resilience against various content-preserving attacks and possess a commendable ability for discrimination.

Subsequently, in 2021, Shaik, Huang, and co-researchers proposed innovative image hashing approaches employing Chromatic channel information [24], [25]. Their methods combine texture with invariant vector distance for feature extraction. While the methods show potential, there is still a necessity for improvement in accurately pinpointing tampered regions and reducing the hash length. Nonetheless, this approach can identify visually similar images that have undergone individual and combined content-preserving manipulations. In the same year, C. Qin, Paul, and their co-researchers presented perceptual image hashing approaches involving convolutional neural networks [26], [27], [28], [29]. The researchers applied a stacked denoising auto-encoder algorithm to retrieve altered codes from tampering. These methods not only identify tampered regions within an image but also aid in restoring the tampered content. However, it cannot autonomously recover color images.

In 2022, a robust image-hashing strategy utilizing Lifting Wavelet Transforms (LWT) and Discrete Cosine Transforms (DCT) [30] was proposed. This approach is based on using LWT-DCT to generate a hash. Although this technique is not invariant to rotations, it detects subtle-level tampering and maintains robustness against non-malicious manipulations.

Similarly, in 2022, Li et al. [31] and Fonseca-Bustos et al. [32] introduced self-supervised learning methods for content identification. These algorithms employ a unified performance evaluation method to create an image hash. However, it has limitations in terms of robustness against geometric operations. Nonetheless, this proposed method can detect content-based image forgery and locate tampered areas.

In 2022, an innovative approach to perceptual image hashing using an autoencoder [33] has been proposed. This algorithm utilizes both the encoder and decoder with convolutional neural networks for its development. This method showcases resilience against different distortions and can identify localized tampering as small as 3% of the original image. However, it does result in a longer hash length [34]. Similarly, in 2023, Yu and their co-authors [35] proposed a speedy and robust image hashing technique employing saliency map features and a sparse model. The method is rooted in the use of two-dimensional principal component analysis, which significantly reduces computational time. Nevertheless, it is sensitive to image rotation. In the same year, Xing H et al. [36] introduced a method focused on Watson's model and locally linear embedding. This approach utilizes the Hu invariant moment, Watson's Model, to create a hash. While this technique excels in copy detection, it does result in a longer hash length.

Similarly, in 2023, Liang et al. [37] proposed an efficient hashing method using 2D-2D PCA (Principal Component Analysis) for Image copy detection. Tang et al. [38] introduced a new robust image hashing with multidimensional scaling. These methods are invariant to rotation, but the hash length is high. Liang et al. [39] implemented robust hashing with Local Tangent Space Alignment for image copy detection. Liang et al. [40] proposed robust hashing via Global and Local Invariant Features for image copy detection. However, the above methods cannot localize the tampered region, which is the main contribution of our work.

The proposed hashing method using ring partition and CS-LBP uses the concept of symmetry instead of just considering the relationship between the central pixel and its neighbors. CS-LBP considers pairs of symmetrically opposite pixels around the central pixel. This symmetric encoding captures texture information that is invariant to certain rotations.

The main advantages of the proposed method is that with this method, it is possible to localize tampering more precisely because of the ''ring partition'' strategy. Center-symmetric LBP features capture local texture patterns, which helps to obtain better robustness against various image transformations compared to other existing image hashing methods. Apart from that, the proposed method is invariant to common image transformations. This method's discriminative texture information helps obtain more accurate tampered region detection and content authentication. The proposed method works on a diverse set of images irrespective of the type of images. This method offers improved accuracy in detecting tampering and content authentication.

III. PROPOSED HASHING ALGORITHM

The fundamental block diagram of the proposed hashing approach is shown in Fig.1. It comprises pre-processing, ring partitioning, feature extraction utilizing CSLBP, and hash generation.

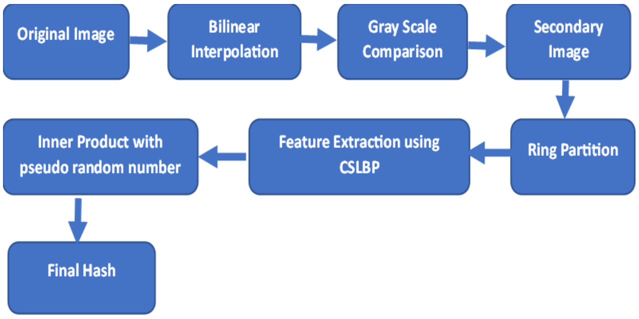


FIGURE 1. Proposed image hashing based on ring partition and CSLBP.

A. PREPROCESSING

At the beginning of this procedure, RGB images undergo the process of traditional color space conversion to be converted into grayscale images. All input images are then uniformly resized to a standard resolution of 512 × 512 using bi-linear interpolation. This standardization is essential to ensure a consistent hash length in the final output, given the variability in the sizes of actual images. Subsequently, Gaussian low-pass filtering is employed on the resized image to mitigate the influence of subsequent minor modifications, such as noise interference or filtering.

B. FEATURE EXTRACTION

Extracting features are segmented into ring partitioning and center symmetric local binary pattern (CSLBP) analysis. To extract features, the image is initially partitioned into concentric rings, and subsequently, center symmetric local binary patterns are employed to capture the characteristics of each ring. In the end, the extracted characteristics are transformed into a string of actual integers, which makes up the final hash value.

C. RING PARTITION

During the ring partition process, the image is partitioned into groups of rings, ensuring that each ring contains an equal area, resulting in a consistent number of pixels across all image rings [11], [19]. Each ring set is employed to construct feature vectors that contribute to the formation of the hash.

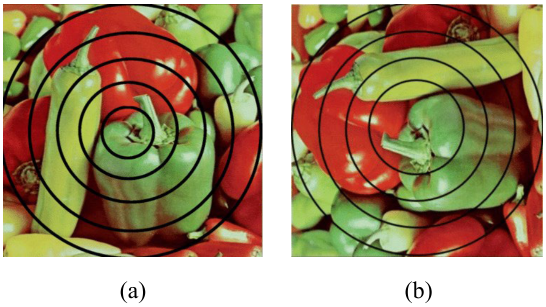


FIGURE 2. Ring partition of the image and its rotated versions.

Fig.2. (a) represents the central portion of the image "Peppers," while Fig. 2. (b) is derived from cropping the "Peppers" image after it has been rotated by 90 degrees. Fig. 2(a) and (b) are a perfect match in terms of the content of the images included inside the rings. To put it another way, the information contained in each ring does not change once the image has been rotated. This enables the extraction of characteristics in an image that are stable against rotation.

Here, the image content lie within the normalized image's inscribed circle, which we divide into rings of equal size. This is due to the expectation that each ring feature will be of equal value to other features. The division is accomplished by computing the pixel distances from the image's center and each circle's radius.

Let q (x, y) be the pixel value of the regularized image of size (1 ≤ x ≤ M) and (1 ≤ y ≤ M), n be the number of rings in the image. S_j is the collection of total pixel values corresponding to the j^{th} ring (j=1,2, 3, . . . , n). Each ring pixel is categorized into different sets based on distance from the image center and the radii. The sets are as follows.

$$Subscript\ S_1 = \{q(x,y) \mid d_{x,y} \le r_1\} \tag{1}$$

$$S_j = \left\{q(x,y) \mid r'_{j-1} \le d_{x,y} \le r_j\right\} \tag{2}$$

the Where r_j is the radius of the j^{th} ring and $d_{x,y}$ is the Euclidean distance from the *q (x, y)* to the center of the image (x_c, y_c), which is defined as

$$d_{x,y} = \sqrt{(x - x_c)^2 + (y - y_c)^2} \tag{3}$$

Here $x_c = \frac{M'}{2} + 0.5$ and $y_c = \frac{M'}{2} + 0.5$ if M' is an even number. Otherwise, $x_c = \frac{(M'+1)}{2}$ and $y_c = \frac{(M'+1)}{2}$ for radii. The area A' and average area μ_A of the inscribed circle can be calculated using (4) and (5).

$$A' = \pi r_n^2 \tag{4}$$

$$\mu_A = \left[\frac{A'}{n}\right] \tag{5}$$

The radii of the circle can be computed by (6).

$$r_1 = \sqrt{\frac{\mu_A}{\pi}} \tag{6}$$

Similarly, for the other circle, the following equation can compute radii.

$$r_k = \sqrt{\frac{\mu_A + \pi r_{k-1}^2}{\pi}} \tag{7}$$

The concept of the inscribed circle's area has been employed in generating hashes. Fig. 3(a) shows how this inscribed region has been divided into uniform, concentric rings. In the secondary view, each ring corresponds to a column, shown in Fig. 3(b).

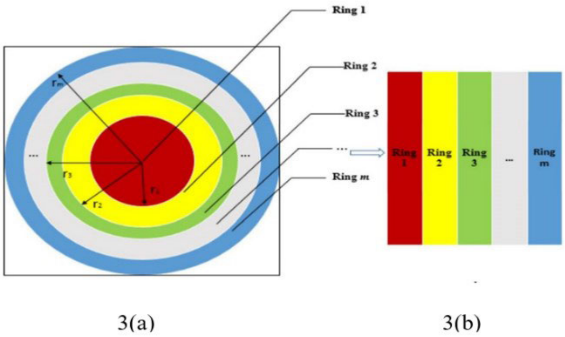
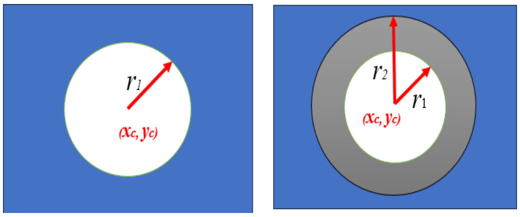


FIGURE 3. Image ring partition and its corresponding column vector.

The initial mask is created with a pixel value of '*1*' (representing 'true') within the circular region defined by the radius r_1. In contrast, the pixel value outside this circle is set to '*0*'



4 (a) 4(b)

FIGURE 4. Mask creation.

(representing 'false'), as depicted in Fig. 4 (a) First mask ($\boldsymbol{T_1}$) for $\boldsymbol{r_1}$, and (b) Second mask ($\boldsymbol{T_2}$) for ($\boldsymbol{r_2} - \boldsymbol{r_1}$)

The very first column of the C matrix can include the values of the first ring r_1 pixels. And this can be achieved by (8).

$$C_1 = I \times T_1\ ;\ inside\ r_1 \tag{8}$$

where I and T_1 are input images and the first mask

Similarly, the second column of the matrix C can be computed by the

$$C_2 = \mathrm{I} \times T_2;\ \text{ for } r_2 - r_1 \tag{9}$$

Similarly, the pixel values for the other regions can be placed in the other columns of matrix C.

$$C_k = \mathrm{I} \times T_k;\ \text{ for } (r_j - r_{j-1}) \tag{10}$$

The rotation invariant matrix is subsequently obtained, as shown in (11)

$$\mathrm{C} = [C_1, C, C_3, \ldots, C_m] \tag{11}$$

CSLBP processes the matrix C to create a hash, which is used to reduce a dimensionality vector.

D. CENTRE SYMMETRIC LOCAL BINARY PATTERN

The process of extracting features from matrix C can be accomplished by utilizing the Centre Symmetric Local Binary Pattern (CSLBP) technique. LBP is a method that encodes the relationship between the central pixel and its surrounding neighbors by comparing their intensity values. It assigns a binary value (0 or 1) to each neighbor pixel based on whether its intensity is greater or lesser than that of the central pixel. These binary values are combined into a binary pattern for various image analysis tasks.

CS-LBP introduces symmetry into the LBP concept. It considers pairs of symmetrically opposite pixels concerning the central pixel and encodes their intensity relationships. This approach helps capture symmetry texture patterns, such as certain fabrics or natural textures.

The LBP operator shown in Fig. 5 is applied to C matrix and can be expressed as:

$$LBP_{P,R}(c) = \sum\nolimits_{i=0}^{P-1} s\left(g_p - g_c\right) 2^i \tag{12}$$

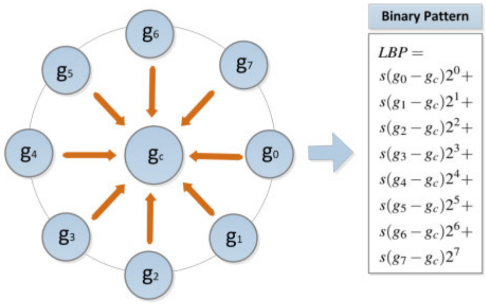


FIGURE 5. LBP descriptor.

where g_c is the center pixel $c = (x_c, y_c)$ is the gray value, g_p is the neighboring pixel gray value, s is a threshold function, and R is the circle's radius.

$$s\left(x\right) = \begin{cases} 1 & if\ x \geq 0 \\ 0 & otherwise. \end{cases} \tag{13}$$

The main drawback of the LBP descriptor is it produces long histograms and is very difficult to use in the context of region descriptors. To overcome this, we use CSLBP, which can capture better gradian information than the LBP. This can be achieved by comparing all neighboring pixel values with center pixel values. We compare and take the difference of the center symmetric pairs of opposite pixels in the neighborhood.

Assuming that $I\left(x, y\right)$ is a grayscale image, let g_c be the grayscale value of the random pixel located at (x_c, y_c), i.e., $g_c = I\left(x_c, y_c\right)$.

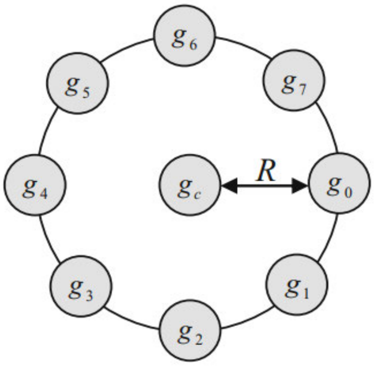


FIGURE 6. CSLBP calculation of 8 neighbor pixels.

Fig. 6 depicts the gray values of neighboring circular pixels that are uniformly spread out throughout an annular circle that has a radius of R, surrounding the center g_c are presented by $g_p, p = 0, 1, \ldots .p - 1$

The CSLBP operator is given by:

$$CSLBP_{P,R,T}\left(x_c, y_c\right) = \sum\nolimits_{i=0}^{\left(\frac{p}{2}\right)-1} s\left(g_p - g_{p+\left(\frac{p}{2}\right)}\right)2^i \tag{14}$$

$$g_p = I\left(x_p, y_p\right), p = 0, 1, \ldots .p - 1$$

$$CSLBP_{P,R,T}\left(g_c\right) = 2^0 \times s\left(g_0 - g_4\right) + 2^1 \times s\left(g_1 - g_5\right) + 2^2 \times s\left(g_2 - g_6\right) + 2^3 \times s\left(g_3 - g_7\right) \tag{15}$$

where p is the total number of neighborhood pixels, g_p, $g_{p+\left(\frac{p}{2}\right)}$ are opposite gray levels of the pixels, and $'s'$ is the threshold function and it is defined as:

$$s\left(x\right) = \begin{cases} 1, & if\ x > T \\ 0, & otherwise. \end{cases} \tag{16}$$

where T is a user-defined threshold. The length of the CSLBP descriptor is given in (17)

$$1 + \sum\nolimits_{i=0}^{\left(\frac{p}{2}\right)-1} 2^i = 2^{p/2} \tag{17}$$

For a central pixel g_c and its evenly spaced circular neighbors g_p, P = 0, 1, 2, 3…P-1. We may measure the gray-level differences within the neighborhood between symmetric center pairs of opposite pixels, as shown in (15).

$$\begin{aligned} & d_{p,q} = g_p - g_q, q = p + \left(P/2\right), \mathrm{P} = 0, 1, \cdots, (P/2 - 1) \\ & d_{p,q} = s_{p,q} \times m'_{p,q} \\ & s_{p,q} = sign\left(g_p - g_q\right), m'_{p,q} = \left|g_p - g_q\right| \end{aligned} \tag{18}$$

where $s_{p,q}$ and $m'_{p,q}$ sign and magnitude of the difference $d_{p,q}$ respectively.

The number of adjacent pixels P, The radius R, and the threshold T are the three variables that comprise the CSLBP operator. The sign function can obtain 16 decimal integers from 0 to 15 for each pixel $I\ (x, y)$ using eight nearby samples, and we discovered that the best results are obtained for values of $P = 8$,$R = 1$,$T = 0.1$.

$$CSLBP_{8,1,0.1}\left(x_c, y_c\right) = \sum\nolimits_{p=0}^{3} s\left(g_p - g_{p+4}\right)2^8 \tag{19}$$

$$MV\ (i, j) = \left[mv'_{(0,4)}, mv_{(1,5)}, mv_{(2,6)}, mv_{(3,7)}\right] \tag{20}$$

Where MV- Magnitude Vector

E. HASH GENERATION

Each non-overlapping block in the initial grayscale image extracts CSLBP features in the proposed image hashing. The Four histograms are constructed considering four vector magnitude components. Histogram computation can be done by using the (21).

$$H_P\left(b\right) = \sum\nolimits_{i=1}^{B} \sum\nolimits_{j=1}^{B} m_{p,q}\left(i, j\right) \times f\left(CSLBP\left(i, j\right), b\right)$$
$$b \in [1, 15], p = 0, 1, 2, 3\ and\ q = p + 4 \tag{21}$$

Here four different histograms are created using this process. The final histogram is obtained by appending the four histograms of an image block, known as Feature Vector FV, which is used for hash generation, and the size of the Feature Vector is 64.

$$FV = [H_1, H_2, H_3, H_4] \tag{22}$$

Pseudo-random weights are generated using $w = \{\alpha_i\}$, $i = 1, \ldots,64$ from the normal distribution $N\left(u, \sigma^2\right)$. The random vector w has the same dimension as each ring's feature vector FV. The inner product of the feature vector and pseudo-random vectors' inner product are used to get the final hash value.

$$h = [FV, w] \tag{23}$$

F. SIMILARITY MEASUREMENTS

Similarity between hashes was determined using the correlation coefficient as a metric and given in (24), shown at the bottom of the next page. High value of hash correlation refers that the visual content of images is visually similar, and a lower value indicates that the images have very different content.

Let $H^{(1)} = (h)_1^{(1)}, (h)_2^{(1)}, (h)_3^{(1)}, \ldots\ldots, (h)_L^{(1)}$ and $H^{(2)} = (h)_1^{(2)}, (h)_2^{(2)}, (h)_3^{2}, \ldots\ldots, (h)_L^{2}$ are two hash vectors of size L, each representing a image. The correlation coefficient is formally defined as (24).

The mean of the two image hash vectors are $\mu^{(1)}$ and $\mu^{(2)}$.

G. TAMPERING DETECTION AND LOCALIZATION

Tampering of an image is an addition or removal of some part in the image used to add or remove some information from the image. The hashing method used in this work can detect image tampering and can also be used to localize the tampered portions of the image.

One of the primary techniques for localizing forged regions is block-based matching. To begin, the image is divided into blocks that either overlap or do not overlap. After that, the perceptual hash method gives each block a hash number. A block-by-block comparison during analysis identifies possible manipulated regions by extracting hashing codes from the appropriate blocks of the suspect image. Block size regulates the trade-off between hash length and detection performance for tampering localization. Smaller hash lengths are produced by larger blocks but may result in more false positives than smaller blocks. For detecting tampering, the hash correlation between each block's portions is found. If the correlation value is less than that of the threshold, then it is classified as a tampered block. A block diagram for the localization of tampered images is given in Fig. 7. The blocks having tampering are distinguished by a black boundary around them.

H. GEOMETRIC CORRECTION

To authenticate the image, an original image and its hash value will be received. If the image undergoes a RST transformation process. The transformation influence has to be eliminated to authenticate the image, which is shown in Fig.8. The RST correction gets affected by tampering, which affects the prior works performance. The composite RST operation on an image is shown in Fig.8. The coordinates with non-zero pixels at the right-most, left-most, bottom-most, and top-most are computed. The following section discuss about resolving the ambiguity of the rotation direction. If an image is rotated at an angle of 30 degrees in an anticlockwise direction, the non-zero pixels at the top-most are greater

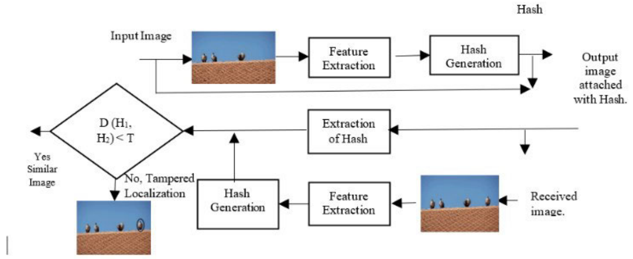


FIGURE 7. Image tamper detection and localization.

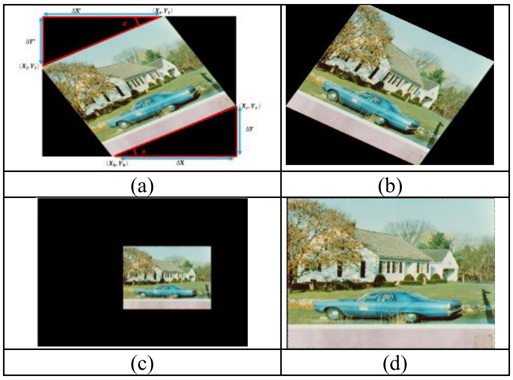


FIGURE 8. Geometric correction (a) Anticlockwise rotation (b) Clockwise rotation (c) Anti rotated image (d) Cropped and resized image.

than the bottom-most coordinate points, i.e., $X_t > X_b$ ($\theta^p \cong \theta'$), otherwise ($\theta^n = \theta'$), rotated in the clockwise direction. Then the image is anti-rotated in θ' angle.

$$\theta = arctan\left(\frac{\Delta Y}{\Delta X}\right), \theta' = arctan\left(\frac{\Delta Y'}{\Delta X'}\right) \tag{25}$$

$\Delta Y = Y_b = Y_r, \Delta X = X_r = X_b, \Delta Y' = Y_l = Y_t, \Delta X' = X_t = X_l$ where $(X_l, Y_l), (X_r, Y_r), (X_t, Y_t), (X_b, Y_b)$ are the indexes of non-zero pixel values of left-most, right-most, top-most, and bottom-most points respectively. Finally, the interested area is cropped and resized based on the dimensions of the input image.

IV. EXPERIMENTS AND RESULTS ANALYSIS

The experimental section primarily consists of analysis of different parameters like robustness, discrimination, key-dependent security tests, experiments to compare the performance of various schemes, copy detection tests, and tamper detection tests. All experiments were executed on the MATLAB R2022b platform, outfitted with an Intel(R) Core (TM) i3-6006U CPU running at 2.50 GHz, 2.5 GHz, and 12 GB RAM.

The proposed hashing algorithm is tested based on its capability to discriminate and its robustness to common content preserving operations. For testing discrimination, different image pairs have been collected, and hash correlation values are obtained between the hashes. A histogram of these values is plotted, allowing us to know the distribution of correlation coefficients for pairs of different images. Another histogram is also plotted to check the robustness of the hashing scheme. We have conducted experiments on standard benchmark images collected from various datasets as shown in Fig.9.

(a) Standard Benchmark images

(b) Images collected from USC-SIPI Image Database

(c) Images collected from Ground Truth Image Database

FIGURE 9. Standard benchmark images collected from various databases.

A. GENERATING HASH AND FINDING CORRELATION

The proposed hashing algorithm is adopted for hash generation. Here, the hash value for an aero-plane image is computed. The same image is then rotated by 45 degrees, its hash is generated, and the correlation value of the original and rotated version is evaluated. This demonstrates a strong correlation between them.

In addition, we compare the hash of this image with another image and then estimate the correlation coefficients once more. This part demonstrates how the hash function works. The following sections perform tests for discrimination and robustness. Fig. 10 shows a high correlation value of similar and a low correlation of dissimilar image pairs. Hence Our proposed technique performs well for discrimination and robustness.

B. EVALUATION OF PERCEPTUAL ROBUSTNESS AND DISCRIMINATION

The suggested model is used to separate the received images into "perceptually similar image pairs," "Tampered image

$$S = \frac{[(H^{(1)} - \mu^{(1)}).(H^{(2)} - \mu^{(2)})']}{\sqrt{[(H^{(1)} - \mu^{(1)}).(H^{(1)} - \mu^{(1)})']} \times \sqrt{[(H^{(2)} - \mu^{(2)}).(H^{(2)} - \mu^{(2)})']}}$$

$$\mu^{(1)} = \frac{1}{L}\sum_{i=1}^{L} h_i^{(1)}, \mu^{(2)} = \frac{1}{L}\sum\nolimits_{i=1}^{L} h_i^{(2)} \tag{24}$$

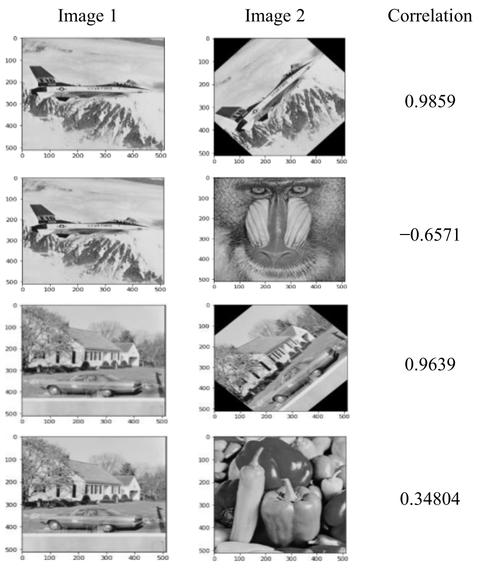


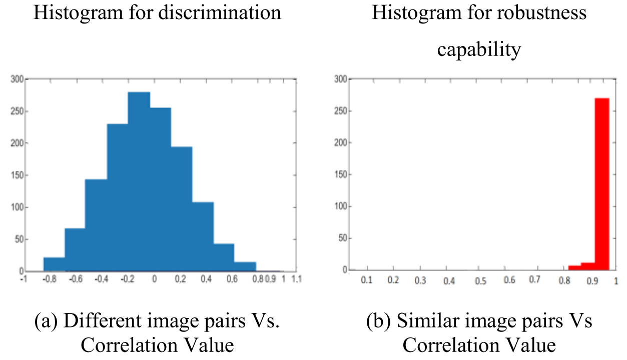
FIGURE 10. Correlation among similar and dissimilar image pairs.

TABLE 1. Content preserving operations.

Operation	Parameter	Parameter Values
Gamma Correction	Gamma	0.75, 0.85, 1.2, 1.3
3x3 Gaussian LPF	Standard deviation	0.3,0.4,0.5, …, 0.9 ,1
Scaling	Ratio	0.5,0.75,0.9,1.1,1.5,2.0
JPEG Compression	Quality	30, 40, ….90, 100
Salt and pepper noise	Density	0.002, 0.004…, 0.01
Speckle noise	Variance	0.001, 0.002, 0.003, .. 0.01
Contrast Change	Adjustment	-20, -10, 10, 20
Brightness Change	Adjustment	-20, -10, 10, 20
Rotation	Angle	2, 4, 6, 8, 10
Watermark embedding	Strength	20,40,60,80,100

pairs,” or “different image pairs”. The threshold correlation values between similar and distinct image pairs are investigated to attain the suggested hashing methods’ perceptual robustness and discriminative capabilities.

A total of 3120 pairs of images with high perceptual similarity, produced by different CPOs, have been used in the experiment. Here, Specifically, 37 ‘Aerial’ and 5 ‘Miscellaneous’ different color images are chosen from the USC-SIPI [42], and ten images are chosen from the CASIA [29] databases of size 512 × 512 to 1024 × 1024, generating a database of 52 × 60 = 3120 pairs of visually identical images using various image operations like changing brightness, contrast, rotation by small angles, etc. listed in the Table 1. In addition, 200 different images are selected from the Ground Truth [41], NITS [43] database, and the Internet. Thus, this creates a database of 200 × (200-1)/2 = 19900 different image pairs. Similarly, 480 tampered image pairings are chosen from CASIA V2.0 [29], where the sizes vary from 512 × 512 to 1024 × 1024 and measured the hash correlation for both the different pairs of images and perceptually identical pairs of images from (24).



(a) Different image pairs Vs. Correlation Value

(b) Similar image pairs Vs Correlation Value

FIGURE 11. Histogram of dissimilar and similar image pairs of the proposed method.

The proposed method’s discrimination capability and perceptual robustness have been evaluated based on the histogram drawn between different, similar pairs and their hash correlations, as shown in Fig. 11.

C. THRESHOLD ESTIMATION

From Fig.11, the threshold value is set at the meeting point of similar and different image pairs so that below that, cutoff images are considered to be different, and above that, they are considered the same or similar images. It is observed from the above figure that the correlation threshold (T) is set at the meeting point of similar and different image pairs, i.e., $0.8 \leq T \leq 0.9$.

FIGURE 12. Original images collected from the CASIA database.

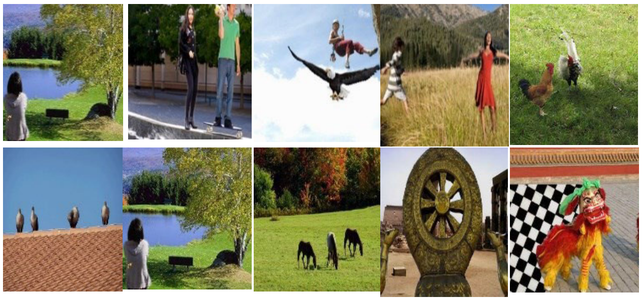


FIGURE 13. Tampered images collected from the CASIA database.

D. TAMPERING DETECTION AND LOCALIZATION

The sensitivity to forgery in images was explored through 480 tampered image pairs selected from the CASIA 2.0 [29]

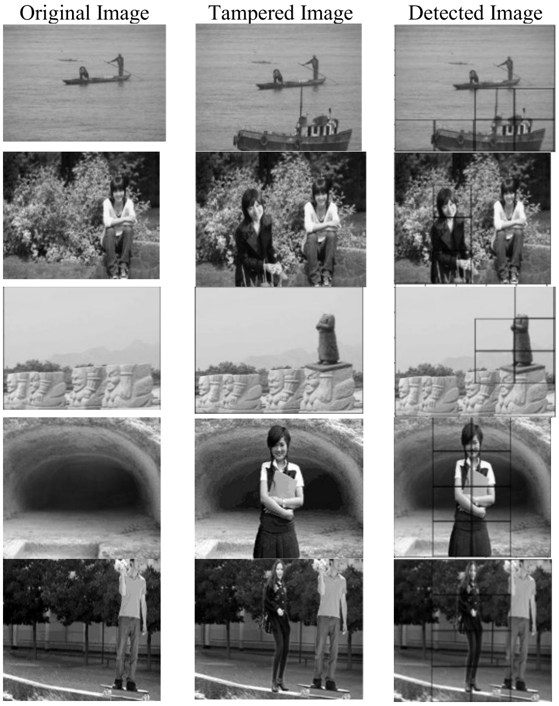


FIGURE 14. Tampering localization performance.

database, where the sizes are from 256 × 256 to 900 × 768 as shown in Fig.12 and Fig.13. The image was divided into blocks to detect image forgery, and the hash correlation between each block of tampered image and the original image was found. If the correlation was found to be below the cutoff, then the block has some amount of tampering. From the Fig.14, it has been found that the majority of the tampered objects may be identified and localized using the proposed method.

E. PERFORMANCE COMPARISONS

In this section, we have conducted a comparative analysis between our proposed approach and several related methods, namely Khelaifi et al. [19], Paul et al. [27], Mengzhu et al. [35], Xing et al. [36]. The benefits and drawbacks of the methods are summarized in Tables 2 to 5. The algorithms' evaluation and comparison have been done on identical datasets [29], [41], [42], [43], the image sizes are from 256 × 256 to 1024 × 1024, and each comparison technique is simulated on the same computer's MATLAB R2022b platform.

The Receiver Operating Characteristics (ROC) curve has been used to evaluate the performances. A ROC curve is plotted using two parameters denoted as (TPR; FPR), where the x-axis denotes False Positive Rate (FPR) and the y-axis signifies True Positive Rate (TPR) shown in (26) and (27). The ROC comparisons and trade-offs between five image hashing algorithms have been demonstrated in Fig.15. The green curve in the top left corner of the graph indicates the proposed method; it is observed that the proposed technique has better discrimination and robustness capability than other existing algorithms. The graph shows that the proposed approach has a larger AUC of 0.9992 than others.

TABLE 2. Robustness comparison (TPR value at an optimal threshold).

Operations	Khelaifi, F et al. [19]	Paul M et al. [27]	Mengzhu et al. [35]	Xing, H. et al. [36]	Proposed method
3 × 3 Gaussian	0.8831	0.8624	0.8453	0.9821	1
Speckle noise	0.9738	0.8737	0.8746	0.7748	1
JPEG compression	0.9318	0.8048	0.9047	0.9242	1
Salt & pepper noise	0.8286	0.8417	0.7438	0.7286	1
Rotation	0.0213	0.0516	0.8564	0.7427	1
Scaling	0.8144	0.8164	0.9143	0.8745	1
Translation	0.0024	0.1741	0.1257	0.8675	1
RST	0.3272	0.0527	0.8317	0.5740	0.9996

TABLE 3. Performance comparisons TPR at an optimal threshold.

Reviewed Techniques	Khelaifi, F et al. [19]	Paul M et al. [27]	Mengzhu et al. [35]	Xing, H. et al. [36]	Proposed
AUC	0.9914	0.9973	0.9998	0.9997	1
Optimal FPR when TPR = 1	3.12×10^{-6}	0.1473	3.22×10^{-5}	0.0017	0.0521
Average Time (s)	1.556	0.2134	17.96	0.1992	0.2234
Optimal TPR when FPR = 0	0.9998	0.8162	0.998	0.9994	0.9999
Execution Time (s)	0.112	0.62	1.1	0.67	0.5

$$TPR = \frac{Total\ Number\ of\ similar\ image\ pairs\ considered\ similar}{Visually\ similar\ image\ pairs} \tag{26}$$

$$FPR = \frac{Total\ Number\ of\ different\ image\ pairs\ considered\ similar}{Visually\ different\ image\ pairs} \tag{27}$$

The performance of the five compared methods against the various CPOs is shown in Table 2. The optimal method will return a high TPR value (close to one) and a low FPR value (close to zero). The proposed technique is observed to have a higher TPR in terms of robustness performance. Especially in the proposed technique, the TPR rate betters that of the compared methods, indicating robustness against brightness and contrast adjustments, JPEG compression, salt, and pepper noise addition, embedding, watermarking, and more. However, it does exhibit some limitations about RST.

The performance comparison at an optimal threshold value has been made using additional factors shown in Table 3.

TABLE 4. Comparison of false positive rates.

Algorithm operations	R. K. K, et al. [10]	R. H. L et al. [15]	Paul M et al. [27]	Abdul. et al. [33]	Proposed method
Severe Tampering	0.4023	0.2486	0.1940	0.0535	0.0072
Minute Tampering	0.4257	0.2315	0.1548	0.3251	0.0524

TABLE 5. Performance comparisons of existing approaches.

Algorithm operations	Khelaifi, F et al. [19]	Paul M et al. [27]	Mengzhu et al. [35]	Xing, H. et al. [36]	Proposed method
Tiny, tampered detection	No	Yes	No	No	Yes
Tampered Localization	No	Yes	No	No	Yes
Hash length	16 digits	1024 digits	640 digits	1024 digits	64 digits
Methods used	CSLBP	Auto encoder	Salient Map	Watsons model, LLE	Ring partition CSLBP
Threshold Value	4	0.98	0.95	0.20	0.8
Performance metric	HD	CC	CC	HD	CC

HD- Hamming Distance, CC- Correlation Coefficient

It is examined that the proposed approach has a higher TPR, i.e., 0.9999, which is higher than other methods, and a lower FPR, i.e., 0.0521, which is inferior to different algorithms. The existing work [35] demonstrates high robustness against content-preserving operations, except for rotation. Our proposed technique focuses on enhancing rotation robustness for slight angle variations. The utilization of the Ring partition technique with CSLBP effectively detects small-scale tampering. Notably, the False Positive Rate (FPR) for minor area tampering is greater than that for larger areas, potentially due to the reduced visibility of small content in the hash.

The performance comparison for tampered pairs' FPR values is shown in Table 4. It is observed that the proposed method has better FPR values for large tampering and small tampering, i.e., 0.0072 and 0.0524.

All image hashing processes were executed on a laptop with an Intel Core i3 processor clocked at 2.5 GHz and 16 GB of RAM, using MATLAB R2022b. The average processing time, estimated from generating hashes for 200 images, is provided in Table 3.

The performance of the existing methods is evaluated with some additional factors, as shown in Table 5. It can be observed that the proposed scheme has a smaller hash length than other techniques. The method [19] also has a small hash length, but it cannot detect and localize the tampered region, which is the major finding in our proposed method. Due to the incorporation of Ring partition and CSLBP techniques, the proposed method shows better performance for robustness and discrimination than other methods.

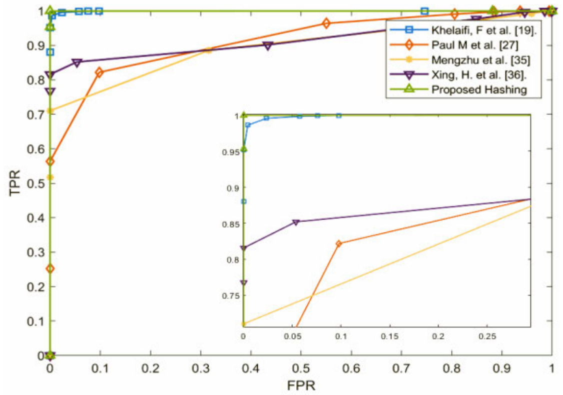


FIGURE 15. ROC comparisons between the proposed strategy and some of the other approaches for robustness and discrimination.

The "ring partition" strategy divides the image into concentric regions, which allows a more focused analysis of different parts of the image. By isolating these regions, the method can more accurately pinpoint where tampering has taken place. By focusing on localized partitions, the proposed method can significantly reduce the amount of data that needs to be processed and analyzed, resulting in faster computation.

V. CONCLUSION

In this work, we have proposed perceptual image hashing using ring partition and CSLBP. The image is converted to a standard image, from which ring-based statistical features are extracted using CSLBP. These features are stable and rotation-invariant. This is accomplished because the ring partition does not affect image rotation. The results have shown that proposed hashing can resist geometric operations to images, including rotation, and offers better performance for robustness and discrimination than others. This method develops desirable discriminative capacity and is sensitive to changes in visual information. It can detect and localize small tampering areas, which is the major drawback of the other existing approaches. Improvement in this proposed hashing scheme can be achieved by reducing the hash length. This is the major limitation of the proposed method.

REFERENCES

[1] Z. Tang, S. Wang, X. Zhang, W. Wei, and S. Su, "Robust image hashing for tamper detection using non-negative matrix factorization," *J. Ubiquitous Converg. Technol.*, vol. 2, no. 1, pp. 18–26, Aug. 2008.

[2] X. Lv and Z. Jane Wang, "Reduced-reference image quality assessment based on perceptual image hashing," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Cairo, Egypt, Nov. 2009, pp. 4361–4364.

[3] Z. Tang, S. Wang, X. Zhang, W. Wei, and Y. Zhao, "Lexicographical framework for image hashing with implementation based on DCT and NMF," *Multimedia Tools Appl.*, vol. 52, nos. 2–3, pp. 325–345, Apr. 2011.

[4] Y. Zhao, S. Wang, X. Zhang, and H. Yao, "Robust hashing for image authentication using Zernike moments and local features," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 1, pp. 55–63, Jan. 2013.

[5] X. Lv and Z. J. Wang, "Perceptual image hashing based on shape contexts and local feature points," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1081–1093, Jun. 2012.

[6] Z. Tang, Y. Dai, and X. Zhang, "Perceptual hashing for color images using invariant moments," *Appl. Math. Inf. Sci.*, vol. 6, no. 2S, pp. 643S–650S, Apr. 2012.

[7] Z. Tang, X. Zhang, L. Huang, and Y. Dai, "Robust image hashing using ring-based entropies," *Signal Process.*, vol. 93, no. 7, pp. 2061–2069, Jul. 2013.

[8] Z. Tang, X. Zhang, and S. Zhang, "Robust perceptual image hashing based on ring partition and NMF," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 3, pp. 711–724, Mar. 2014.

[9] R. K. Karsh and R. H. Laskar, "Perceptual robust and secure image hashing using ring partition-PGNMF," in *Proc. IEEE Region 10 Conf. (TENCON)*, Macau, China, Nov. 2015, pp. 1–6.

[10] R. K. Karsh, R. H. Laskar, and B. B. Richhariya, "Robust image hashing using ring partition-PGNMF and local features," *SpringerPlus*, vol. 5, no. 1, p. 1995, Dec. 2016.

[11] Z. Tang, X. Zhang, X. Li, and S. Zhang, "Robust image hashing with ring partition and invariant vector distance," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 1, pp. 200–214, Jan. 2016.

[12] R. Davarzani, S. Mozaffari, and K. Yaghmaie, "Perceptual image hashing using center-symmetric local binary patterns," *Multimedia Tools Appl.*, vol. 75, no. 8, pp. 4639–4667, Apr. 2016.

[13] R. K. Karsh, R. H. Laskar, and Aditi, "Robust image hashing through DWT-SVD and spectral residual method," *EURASIP J. Image Video Process.*, vol. 2017, no. 1, p. 31, Dec. 2017.

[14] A. Saikia, R. K. Karsh, and R. H. Lashkar, "Image authentication under geometric attacks via concentric square partition based image hashing," in *Proc. IEEE Region 10 Conf. (TENCON)*, Penang, Malaysia, Nov. 2017, pp. 2214–2219.

[15] R. K. Karsh, A. Saikia, and R. H. Laskar, "Image authentication based on robust image hashing with geometric correction," *Multimedia Tools Appl.*, vol. 77, no. 19, pp. 25409–25429, Oct. 2018.

[16] Z. Tang, L. Chen, X. Zhang, and S. Zhang, "Robust image hashing with tensor decomposition," *IEEE Trans. Knowl. Data Engineering.*, vol. 31, no. 3, pp. 549–560, Mar. 2019.

[17] C. Qin, Y. Hu, H. Yao, X. Duan, and L. Gao, "Perceptual image hashing based on Weber local binary pattern and color angle representation," *IEEE Access*, vol. 7, pp. 45460–45471, 2019, doi: 10.1109/ACCESS.2019.2908029.

[18] X. Wang, X. Zhou, Q. Zhang, B. Xu, and J. Xue, "Image alignment based perceptual image hash for content authentication," *Signal Process., Image Commun.*, vol. 80, Feb. 2020, Art. no. 115642, doi: 10.1016/j.image.2019.115642.

[19] F. Khelaifi and H. He, "Perceptual image hashing based on structural fractal features of image coding and ring partition," *Multimedia Tools Appl.*, vol. 79, nos. 27–28, pp. 19025–19044, Jul. 2020, doi: 10.1007/s11042-020-08619-w.

[20] S. M. Abdullahi, H. Wang, and T. Li, "Fractal coding-based robust and alignment-free fingerprint image hashing," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 2587–2601, 2020, doi: 10.1109/TIFS.2020.2971142.

[21] Z. Tang, H. Zhang, S. Lu, H. Yao, and X. Zhang, "Robust image hashing with compressed sensing and ordinal measures," *EURASIP J. Image Video Process.*, vol. 2020, no. 1, Dec. 2020, Art. no. 21, doi: 10.1186/s13640-020-00509-3.

[22] Y. Zhao and X. Yuan, "Perceptual image hashing based on color structure and intensity gradient," *IEEE Access*, vol. 8, pp. 26041–26053, 2020, doi: 10.1109/ACCESS.2020.2970757.

[23] H. Hamid, F. Ahmed, and J. Ahmad, "Robust image hashing scheme using Laplacian pyramids," *Comput. Electr. Eng.*, vol. 84, Jun. 2020, Art. no. 106648, doi: 10.1016/j.compeleceng.2020.106648.

[24] A. S. Shaik, R. K. Karsh, and M. Islam, "Robust image hashing using chromatic channel," in *Proceeding of Fifth International Conference on Microelectronics, Computing and Communication Systems* (Lecture Notes in Electrical Engineering), vol. 748. Singapore: Springer, 2021, doi: 10.1007/978-981-16-0275-7_5.

[25] Z. Huang and S. Liu, "Perceptual image hashing with texture and invariant vector distance for copy detection," *IEEE Trans. Multimedia*, vol. 23, pp. 1516–1529, 2021, doi: 10.1109/TMM.2020.2999188.

[26] C. Qin, E. Liu, G. Feng, and X. Zhang, "Perceptual image hashing for content authentication based on convolutional neural network with multiple constraints," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 11, pp. 4523–4537, Nov. 2021, doi: 10.1109/TCSVT.2020.3047142.

[27] M. Paul, A. J. Thakuria, R. K. Karsh, and F. A. Talukdar, "Robust color image hashing using convolutional stacked denoising auto-encoders for image authentication," *Neural Comput. Appl.*, vol. 33, no. 20, pp. 13317–13331, Oct. 2021, doi: 10.1007/s00521-021-05956-1.

[28] A. S. Shaik, R. K. Karsh, M. Islam, and R. H. Laskar, "A review of hashing based image authentication techniques," *Multimedia Tools Appl.*, vol. 81, no. 2, pp. 2489–2516, Jan. 2022, doi: 10.1007/s11042-021-11649-7.

[29] *CASIA Tampered Image Detection Evaluation Database*. Accessed: Jan. 27, 2023. [Online]. Available: http://forensics.idealtest.org/

[30] A. S. Shaik, R. K. Karsh, M. Suresh, and V. K. Gunjan, "LWT-DCT based image hashing for tampering localization via blind geometric correction," in *ICDSMLA 2020* (Lecture Notes in Electrical Engineering), vol. 783. Singapore: Springer, 2022, doi: 10.1007/978-981-16-3690-5_156.

[31] X. Li, C. Qin, Z. Wang, Z. Qian, and X. Zhang, "Unified performance evaluation method for perceptual image hashing," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 1404–1419, 2022, doi: 10.1109/TIFS.2022.3161149.

[32] J. Fonseca-Bustos, K. A. Ramírez-Gutiérrez, and C. Feregrino-Uribe, "Robust image hashing for content identification through contrastive self-supervised learning," *Neural Netw.*, vol. 156, pp. 81–94, Dec. 2022, doi: 10.1016/j.neunet.2022.09.028.

[33] A. S. Shaik, R. K. Karsh, M. Islam, and S. P. Singh, "A secure and robust autoencoder-based perceptual image hashing for image authentication," *Wireless Commun. Mobile Comput.*, vol. 2022, pp. 1–17, Oct. 2022, doi: 10.1155/2022/1645658.

[34] S. P. Singh, G. Bhatnagar, and A. K. Singh, "A new robust reference image hashing system," *IEEE Trans. Depend. Sec. Comput.*, vol. 19, no. 4, pp. 2211–2225, Jul. 2022, doi: 10.1109/TDSC.2021.3050435.

[35] M. Yu, Z. Tang, Z. Li, X. Liang, and X. Zhang, "Robust image hashing with saliency map and sparse model," *Comput. J.*, vol. 66, no. 5, pp. 1241–1255, May 2023, doi: 10.1093/comjnl/bxac010.

[36] H. Xing, H. Che, Q. Wu, and H. Wang, "Image perceptual hashing for content authentication based on Watson's visual model and LLE," *J. Real-Time Image Process.*, vol. 20, no. 1, p. 7, Feb. 2023.

[37] X. Liang, Z. Tang, Z. Huang, X. Zhang, and S. Zhang, "Efficient hashing method using 2D–2D PCA for image copy detection," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 4, pp. 3765–3778, Apr. 2023, doi: 10.1109/TKDE.2021.3131188.

[38] Z. Tang, Z. Huang, X. Zhang, and H. Lao, "Robust image hashing with multidimensional scaling," *Signal Process.*, vol. 137, pp. 240–250, Aug. 2017, doi: 10.1016/j.sigpro.2017.02.008.

[39] X. Liang, Z. Tang, X. Zhang, M. Yu, and X. Zhang, "Robust hashing with local tangent space alignment for image copy detection," *IEEE Trans. Depend. Sec. Comput.*, early access, Aug. 22, 2023, doi: 10.1109/TDSC.2023.3307403.

[40] X. Liang, Z. Tang, Z. Li, M. Yu, H. Zhang, and X. Zhang, "Robust hashing via global and local invariant features for image copy detection," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 20, no. 1, pp. 1–22, Jan. 2024, doi: 10.1145/3600234.

[41] *Ground Truth Database*. Accessed: Jan. 27, 2023. [Online]. Available: http://imagedatabase.cs.washington.edu/groundtruth/

[42] (2007). *USC-SIPI Image Database*. Accessed: Jan. 27, 2023. [Online]. Available: http://sipi.usc.edu/database/

[43] (2017). *NITS Image Hashing Database*. Accessed: Jan. 27, 2023. [Online]. Available: https://rishabhphukan.wixsite.com/rkkarsh

ABDUL S. SHAIK was born in 1982. He received the B.Tech. and M.Tech. degrees from Jawaharlal Nehru Technological University Hyderabad (JNTUH), in 2005 and 2010, respectively. He is currently pursuing the Ph.D. degree with the National Institute of Technology (NIT) Silchar, Assam, India. He is an Associate Professor with the CMR College of Engineering and Technology, Hyderabad, India. His research interests include robust image hashing, image authentication, and multimedia security.

RAM K. KARSH (Senior Member, IEEE) received the B.Tech. degree from the Government Engineering College, Bilaspur, in 2009, the M.Tech. degree from the National Institute of Technology Patna (NIT Patna), in 2012, and the Ph.D. degree from NIT Silchar, in 2018. He was a Junior Teaching-cum-Research Fellow with the Birla Institute of Technology, Mesra. He is currently an Assistant Professor with NIT Silchar. His research interests include robust image hashing and image authentication.

MOHIUL ISLAM was born in 1988. He received the B.E. degree from the Assam Engineering College, in 2011, the M.Tech. degree from the National Institute of Technology Agartala, in 2014, and the Ph.D. degree from the Department of Electronics and Communication Engineering, National Institute of Technology (NIT) Silchar, Assam, India. He is currently an Assistant Professor with the School of Electronics Engineering, Vellore Institute of Technology (VIT), Vellore, Tamil Nadu, India. His research interests include image processing, machine learning, and multimedia data security.

SHUBHASHISH BHAKTA was born in Ranchi, Jharkhand, India, in 1986. He received the B.E. degree in electrical engineering and the M.Tech. degree in electrical engineering with specialization in instrumentation engineering from the National Institute of Technology (NIT) Agartala, Tripura, India, in 2009 and 2012, respectively, and the Ph.D. degree from the Indian Institute of Technology (Indian School of Mines) Dhanbad, India. He was with public and private universities, India, as an Assistant Professor. He is currently an Assistant Professor (Chief Researcher) with the EPCE Program, Adama Science and Technology University, Ethiopia. His research interests include hybrid renewable power generation and nonlinear control.