Ring Partition and DWT based Perceptual Image Hashing with Application to Indexing and Retrieval of Near-Identical Images

Govindaraj P[1] and Sandeep R[2]
Cambridge Institute of Technology, K. R Puram, Bengaluru, India

Abstract— The perceptual image hashing function maps an input image to a fixed size short binary string called the perceptual hash. The perceptual image hashing has been used for image authentication, image copy detection, image tamper detection and digital watermarking applications. In this paper, the perceptual hash value is generated using ring partition and Discrete Wavelet Transform (DWT). An rotation-invariant secondary image is constructed using ring-partition technique. Then, the DWT is applied on the secondary image using various mother wavelets, to generate a final hash value. The experiments show that the proposed hashing algorithm is robust against content-preserving operations like image rotation and cropping, Gaussian low-pass filtering, JPEG compression, brightness adjustment, contrast adjustment, motion filtering, salt and pepper noise, circular average filtering and gamma correction. Using the proposed algorithm, an application for indexing and retrieval of near-identical images is developed. The R-precision curves are used to evaluate the performance of the proposed method and some of the existing algorithms.

Keywords— *Image hasing; DWT; Ring partition.*

I. INTRODUCTION

Due to fast development in the multimedia field, pocket sized multimedia devices are available. The devices like smart phone, digital cameras are ruling this generation because it is easier to use, easier to carry and the devices are cheaper. By using these devices it is easier to capture, store and share images and videos, which introduces problems in managing large image database. It is not possible to determine if an image is already existing in a database or not, without searching the whole database using direct comparison. The images that have undergone content preserving manipulations such as brightness change, contrast change, noise addition with acceptable limit and so on, convey the same information to the human eye but may have different digital representation. Such images may be called as near-identical images. The images that have undergone content-changing manipulations such as malicious modification, addition/deletion of objects, high percentage of cropping and so on, change the visual information present in the original image. Such images are considered to be perceptually different image from the original image. Moreover, it is difficult to compare two images, which appears as identical to human eye but having different digital representations. The traditional cryptographic hashing techniques are very sensitive to input data and fails to capture the perceptual similarity of images, generating different hash values even for near-identical images. These demands lead to develop new algorithms to generate suitable image identifiers, known as image hashing. Fig. 1.1 shows the requirement of the image authentication with practical example. Fig. 1.1a shows the former US president with First lady. Fig. 1.1b shows the JPEG compressed version of the original image, and Fig. 1.1c shows the tampered version of the original image, malicious change made in lady's face. The hash values extracted from the Fig. 1.1a and Fig. 1.1b are expected to be same and those of Fig. 1.1a and Fig. 1.1c to be significantly different.

(a) Original. (b) JPEG compressed. (c) Tampered.

Figure 1.1 Example to illustrate the requirements of a hash in image authentication.

II. RELATED WORK

Depending upon the techniques, the hashing algorithms are roughly classified as follows:

The first class includes hashing algorithms based on the DWT [1], [4], [5] and [6]. Venkatesan et al. [1] generated image hashes by using statistics of wavelet coefficients. This method was good against JPEG compression, rotation up to 2 degrees, median filtering, contrast and brightness adjustment. But failed against gamma correction. Monga et al. [6] proposed an algorithm to make short image hashes, using visually significant feature point compression. By using DWT and Secure Hash Algorithm (SHA)-1, Ahmad et al. [4] gave a image hashing scheme for authentication, which was good against tamper detection but failed against normal processing such as contrast and brightness changes. The DWT based image hashing algorithm generates the image hash values based on the image statistics vector, which is extracted from the various sub-bands in the wavelet decomposition of image. The image statistics such as, average value of coarse sub-band

and variances of the other sub-bands remains same under large content preserving modifications to the image. Since this work is inspired from the work carried out by Venkatesan et al. in [1], the robust image hashing algorithm of [1] is discussed in detail. The input image is randomized by dividing each sub-bands into random rectangles using a secret key. The image statistics are then extracted and quantized. The quantized values are applied to the decoding stage of a Reed-Muller error correcting code to generate hash values.

The second class of image hashing algorithms make use of discrete cosine transform (DCT) [7], [8]. Fridrich and Goljan [7] proposed a robust hashing algorithm based on DCT coefficients, for application in digital watermarking field. This hashing was robust against normal processing like brightness and contrast adjustment, but not against image rotation. By using invariant relationship between DCT coefficients, Lin and Chang [8] developed an algorithm. This method was not good against JPEG compression and against image rotation. The hash vales of image is based on preserving selected DCT coefficients. The large changes to the low frequency DCT coefficients of the image does not change the appearance of the image significantly. The first step is to generate several smooth and zero mean random patterns, where $i = 1,2,3,....N$. Then, the DCT block B from the image and the patterns are considered as vectors. In the next stage, the image I is projected on each pattern and absolute value is compared with the threshold to get N bits, where,

$$\text{if } \left|B.\mathrm{P}^{(i)}\right| < Th\, b_i = 0 \tag{2.1}$$

$$\text{if } \left|B.\mathrm{P}^{(i)}\right| \geq Th\, b_i = 1 \tag{2.2}$$

The patterns have a zero mean so that the projections does not depend on the mean gray value of the block. It only depends on the variations within the same block. Thus, this class of image hashing method proved that, it is quite effective for conventional image processing applications.

In third class of image hashing algorithm, Radon transform (RT) was used to develop image hashing algorithm [9], [10] and [11]. Lefebvre et Al. [9] are the first to use RT to construct image hashes. Seo et Al. [11] designed an image hashing algorithm by using the autocorrelation of each projection in the RT domain. Roover et Al. [10] designed a hashing algorithm called RASH method. In this method they divided an image into a set of radial projections of the image pixels and extracted the radial variance vector from the radial projections, and compressed it by using DCT. This method was resilient to image re-scaling and image rotation, but failed against some normal digital processing. This method has large application in medical image processing. When the X-Rays pass through an organ in a tomography, its attenuation depends on the content of organ, distance, and direction or angle of this projection. This projections are called as Radon transform. The projections give a unique representation for individual image. In the RT domain, it is required to calculate some invariant points included in a set of fixed length element. Each projection outputs are not unique and depends on the scaling factor and angle of rotation. The medium points of each projection of each angle are invariant and would retain all the properties of RT.

The medium points are calculated using the following equation,

$$p_{middle} = \frac{\sqrt{width^2 + height^2}}{2} \tag{2.3}$$

The hash vector is generated using the above method of extraction on all angles (up to 180° in steps of 1°). This method can be used in pattern recognition to retrieve an image in a database and in watermarking process.

The algorithms in the fourth class are [12], [13], and [14]. Discrete Fourier Transform (DFT) have used to generate image hashes. By using DFT coefficients, Swaminathan et al. [12] produced image hashes. This algorithm was good against several content preserving operations like filtering and geometric transforms. Wu et al. [13] used RT, combining with DFT and DWT to develop a new image hashing algorithm. This algorithm was resilient against print-scan attack and rotation up to 20 degrees. The method mainly contains three steps as explained. In first step, a key dependent feature vector was generated from the image based on the Fourier transform features. In next step, the feature vectors are quantized. In the final step, the quantized vectors was compressed and hash values were generated.

The fifth class is based on matrix factorization. Kozat et al. [15] proposed an algorithm to calculate hash values by using singular value decompositions (SVDs). This method is also known as SVD-SVD hashing. This method was resilient against image rotation. Monga and Mihcak [16] used non-negative matrix factorization (NMF) to produce image hashes. For a given image, sub-images were selected and NMF is applied. The secondary image was constructed using low-rank matrix approximation. Again, the NMF is applied on the secondary image to generate the co-efficient and basis matrix. By concatenating columns of coefficient matrix and rows of basis matrix, the hash values were generated. This method was robust against geometric attacks, but failed against normal manipulations like watermark embedding. Tang et al. [3] introduced a new algorithm, by constructing secondary image based on ring partition and generated hash values by concatenating the coefficient matrix values. This method was robust against normal image processing and rotation attacks.

III. MATHEMATICAL FRAMEWORK

In [3], Tang et al. developed an image hashing algorithm based on ring partition and NMF. The proposed method is based on ring partition and DWT. So, in this section mathematical framework of NMF, the ring partition technique and DWT are explained.

A. Non-negative Matrix Factorization

NMF is used to reduce the dimension. NMF has been successfully used in image analysis, signal separation, image representation, and face recognition and so on. A nonnegative matrix $\mathbf{V} = (V_{i,j})_{M \times P}$ is generally a combination of P vectors of size M × 1. The results of V are two nonnegative matrices, i.e., $\mathbf{B} = (B_{i,j})_{M\times K}$ and $\mathbf{C} = (C_{i,j})_{K\times P}$. R is the rank of NMF and $R < min\ (M, P)$. The matrix C and matrix B are called as coefficient matrix and base matrix respectively. To represent V approximately they can be used as:

$$\mathbf{V} \approx \mathbf{BC} \tag{3.1}$$

In this method multiplicative update rule is used to find C and B as shown below:

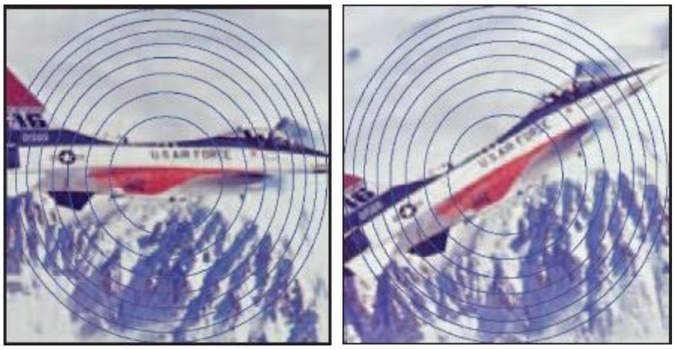
$$B_{i,k} \leftarrow B_{i,k} \frac{\sum_{j=1}^{P} C_{k,j} V_{i,j} / (BC)_{i,j}}{\sum_{j=1}^{P} C_{k,j}} \tag{3.2}$$

$$C_{i,j} \leftarrow C_{k,j} \frac{\sum_{i=1}^{M} B_{i,k} V_{i,j} / (BC)_{i,j}}{\sum_{i=1}^{M} B_{i,k}} \tag{3.3}$$

Where $i = 1, 2, ..., M;\ j = 1, 2, ..., P;\ k = 1, 2, ..., R$.

B. Ring Partition

The rotation manipulation takes image center as origin. If the image is rotated at any angle, the center coordinates remains same. This is the basic idea behind the construction of secondary image. Fig. 3.1a shows the original image, i.e. a central part of Airplane. Fig. 3.1b shows rotated version of the original image. The visual contents are kept unchanged after rotation.



(a) Central part of Airplane. (b) Rotated version.

Figure 3.1: Ring Partition of an image and rotated version.

The schematic diagram of secondary image construction is as shown in Fig. 3.2. The left side indicates that the square image is divided into seven rings and the right side of the Fig. 3.2 shows the construction of secondary image from the square image.

Assume the size of normalized image to be $N \times N$. Let the total number of rings to be n. $\mathbf{R}_k$ be the set of pixel values in the kth ring (k = 1, 2, 3... n). The pixels in the inscribed circle of the image are used. Each column of the secondary image is constructed by each rings. The partition of ring is done by calculating the distance between each pixels and the image center and circle radii. As shown in Fig. 3.2 the pixel values of each ring is calculated by using two neighbor radii, except for the innermost ring. Take r_k as the kth radius (k = 1, 2, 3... n), which is labeled as higher value. r_1 is the radius of the innermost ring and r_n is the radius of outer most ring. It is clear that $r_n = N/2$ for an $N \times N$ image. To find other radii values, the area of the circle A and the average area of individual ring μ_A are calculated as follows, $\lfloor \bullet \rfloor$ means downward rounding.

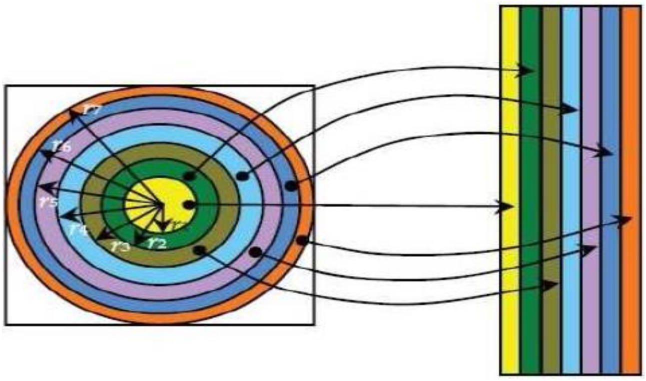


Figure 3.2: Schematic diagram of secondary image construction.

$$A = \pi r_n^2 \tag{3.4}$$

$$\mu_A = \left\lfloor \frac{A}{n} \right\rfloor \tag{3.5}$$

r_1 can calculated by,

$$r_1 = \sqrt{\frac{\mu_A}{\pi}} \tag{3.6}$$

Thus, other values i.e., $r_k (k = 2, 3, 4, ..., n-1)$ can be obtained by the equation given below:

$$r_k = \sqrt{\frac{\mu_A + \pi r_{k-1}^2}{\pi}} \tag{3.7}$$

Let $p(x, y)$ be the pixel value in the xth column and yth row of the image (x_c, y_c). (x_c, y_c) are the coordinates of image center, and are calculated by $x_c = (N+1)/2$ and $y_c = (N+1)/2$, if N odd. Otherwise, $x_c = (N/2) + 0.5$ and $y_c = (N/2) + 0.5$.. The distance between the image center (x_c, y_c) and $p(x, y)$ is calculated by using Euclidean distance as given below:

$$d_{x,y} = \sqrt{(x - x_c)^2 + (y - y_c)^2} \tag{3.8}$$

After calculating the circle radii and distance between pixels, the pixel values can be classified as follows.

$$\mathbf{R}_1 = \{ p(x, y) \mid d_{x,y} \le r_1 \}, \tag{3.9}$$

$$\mathbf{R}_k = \{ p(x, y) \mid r_{k-1} < d_{x,y} \le r_k \} (k = 2, 3, 4, ..., n). \tag{3.10}$$

In next step, rearrange the elements of $\mathbf{R}_k (k = 1, 2, 3, ..., n)$ to make a new sorted vector $\mathbf{u}_k$ in ascending order. This is to ensure that $\mathbf{u}_k$ is unrelated to rotation. The pixel numbers of each set is not always equal to μ_A. The pixels of every ring forms a column of secondary image. $\mathbf{u}_k$ is then mapped into a new vector $\mathbf{v}_k$ by using linear interpolation. The size of $\mathbf{v}_k$ is $\mu_A \times 1$. The secondary image $\mathbf{V}$ is obtained by rearranging new vectors as shown below:

$$\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots\ldots, \mathbf{v}_n] \tag{3.11}$$

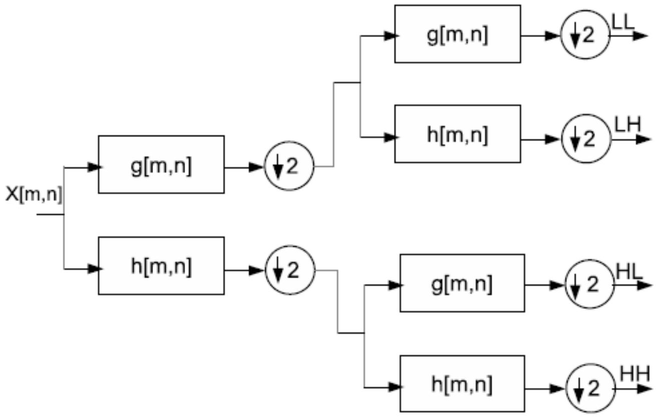
$\mathbf{V}$ is invariant to rotation because $\mathbf{v}_k$ is unrelated to rotation.

C. Discrete Wavelet Transform

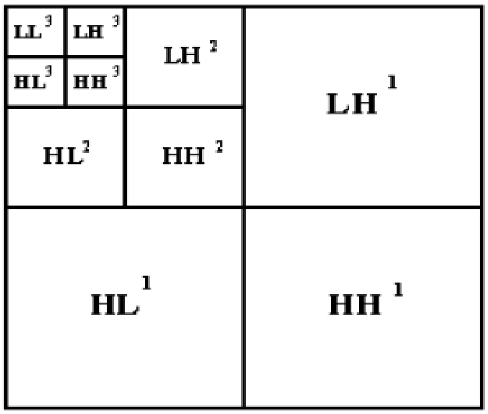
In DWT, the wavelets are discretely sampled. The main advantage of DWT is that, it captures both location information and frequency. The level-3 wavelet decomposition is shown in Fig.3.3. The purpose of the filters $h[m,n]$ and $g[m,n]$ is to extract high frequency components and low frequency components from the input $x[m,n]$. By using ideal filters, by using ideal filters, perfect signal decomposition can be archived. In the first level of decomposition the input image is divided into four components, namely,

- The low frequency components (LL).
- The high frequency components in X direction (LH).
- The high frequency component in Y direction (HL).
- The high frequency component in X − Y direction (HH).

As the decomposition level increases, same process continues on the LL sub-band obtained from the previous level. The LL sub-band contains important information and degree of importance decreases to the other sub-bands. HH sub-band contains least information.



(a) Sub-band decomposition.



(b) Third level wavelet decomposition.

Figure 3.3: Wavelet decomposition.

IV. PROPOSED ALGORITHM

A. Block diagram

The block diagram of proposed algorithm is as shown in Fig. 4.1. It mainly consists of three steps namely,

1. Preprocessing.
2. Ring partition.
3. Discrete Wavelet Transform.

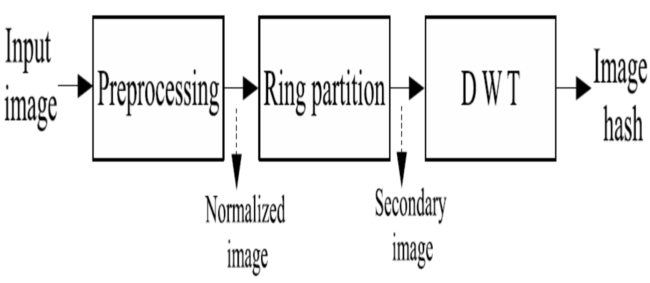


Figure 4.1: Block diagram of proposed algorithm.

B. Preprocessing

Input image is normalized to a size of $N \times N$. This ensures the hashing method has uniform hash length for different sized images and the method is resilient to scaling operation. An RGB color image is converted into YCbCr color space by using the following relation:

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37,797 & -74.203 & 112 \\ 112 & -93.783 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \tag{4.1}$$

where R, G and B denotes red, green and blue components, respectively. Y denotes the luminance, Cb and Cr represent blue difference chroma and red difference chroma, respectively. After conversion, the Y i.e. luminance component is used to represent the image.

C. Ring partition

The secondary image of the original image and the image to be compared will be same after constructing rotation invariant secondary image. The rotation manipulation alters the pixel values of the image, other than center coordinates. The normalized square image (size $N \times N$) is divided into n rings. Then, depending on the radii of the rings, the secondary image is constructed. The construction of secondary image $\mathbf{V}$ is explained in detail in Section 3.

D. Discrete Wavelet Transform

The wavelet decomposition of the secondary image is computed. Each sub-band is randomly tiled into rectangles of small size. Then, the statistics of each rectangle, such as, averages of coefficients of the rectangles in the coarse sub-band and variances of the other sub-bands is calculated. Later, using randomized rounding the values are quantized. The steps of the algorithm to construct the hash value from the secondary image is explained below.

Firstly, the random tiling transform and statistic calculations are performed. The input image I is randomly tiled using a secret key K, as shown in Fig. 4.2.

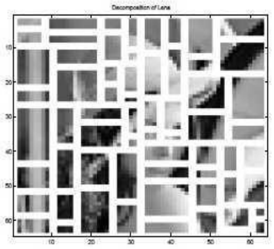


Figure 4.2: Random tiling of Lena's coarse sub-band.

This is randomized analog method for computing integrals through approximation. The large number of samples ensure that there is more number of information about original image, when it under goes some unpredictable and irreversible process. The average value of coefficients in the rectangles are calculated for coarse sub-band. The variance values are calculated for other sub-bands. The length l statistics of image is obtained, and it is mapped as $m = \sigma(I, K)$, where σ is the statistics of the mapping operation. Secondly, the randomized rounding is applied on vector m using a randomized quantizer. This helps the rounded statistics to increase robustness against various attacks. The randomized rounding is the crucial source of entropy in the output of hash function. The quantizer function uses pseudo-random seed from previous step to generate probabilistic distribution, which is used for quantization. This step produces a length l vector.

$$x = Q(m, K) \in \{0,1,2,....,7\}^l . \quad (4.2)$$

V. EXPERIMENTAL RESULTS

The experiments were carried out using MATLAB. The test images in the database was downloaded from [19] and few images are captured using digital camera. The image database consists of 1632 images, which are attacked versions of 40 natural images. For example, an original image and its tampered versions are shown in Fig. 5.1. The level 3 wavelet decomposition is used. The wavelets used are, Haar, Daubechies, Symlets, Coiflets, BiorSplines and ReverseBior.

The input images are resized to 512×512. Number of rings used are 32. The attacks considered for testing the performance of the algorithm are bright-ness modification and contrast modification rotation and cropping, filtering, and noise addition. The images were attacked using the following tools: Scripjpg, PhotoELF and MATLAB. Table I gives the content-preserving operations performed on the image and their parameter values.

The threshold values between the original image and its tampered versions are given in Table II. The results of two different images are given in Table III.

(a) Original. (b) Brightness. (c) Circular filter.

(d) Compressed. (e) Contrast. (f) Motion filter.

(g) Gamma correction. (h) Gaussian filter. (i) Salt and pepper noise.

(j) Rotated. (k) Different image.

Figure 5.1: Image, its tampered versions and different image.

A. *Application*

An application to index and retrieve near-identical images is developed using the image hash generated from the proposed algorithm. The block diagram of the application is shown in the Fig. 5.3. A master database of 150 images was created from 15 images originally present in the database using content preserving manipulations. For example, for a given image, the nine variations of the original image is created (using brightness adjustment, contrast adjustment, gamma correction, filtering, salt and pepper noise, compression and rotation manipulations etc.). The hash value was generated for each of the image and the corresponding meta data was added. The query to the image retrieval system is presented in the form of query-by-example.

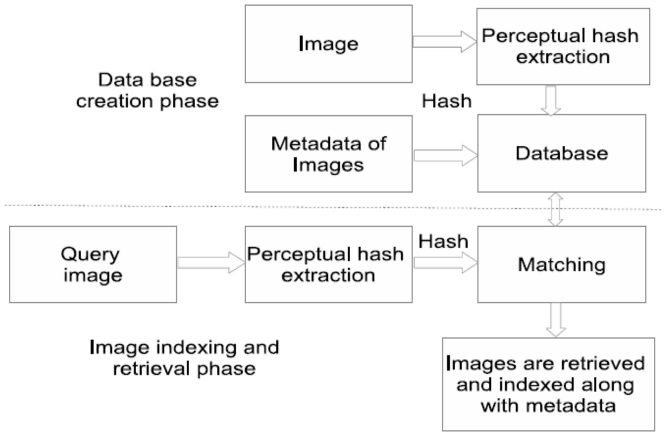


Figure 5.3: Block diagram of perceptual hashing based image indexing and retrieval application.

The hash value is computed for the query image and then matched with the hash values in the database. The near-identical images are then retrieved according to the match between the hash values for the query image and the hash values in the database. The hash for the image was first generated using the proposed algorithm and then the performance of the image retrieval system was evaluated.

The performance of the image retrieval system is evaluated using the R-precision curves [20]. The R-precision curve is the precision at the R-th position in the ranking of results for a query that has R near-identical images. Precision is the ratio of the number of relevant images retrieved to the total number of images retrieved. The precision value indicates the ability to retrieve top ranked images that are near-identical. The curve closest to the upper right-hand corner of the graph indicates the better performance.

Fig. 5.4 shows the comparison of the proposed method and few existing method. From the figure, it can be observed that the proposed algorithm has better performance compared to the hashing algorithm by Tang. et al. in [3] for indexing and retrieval of near-identical images.

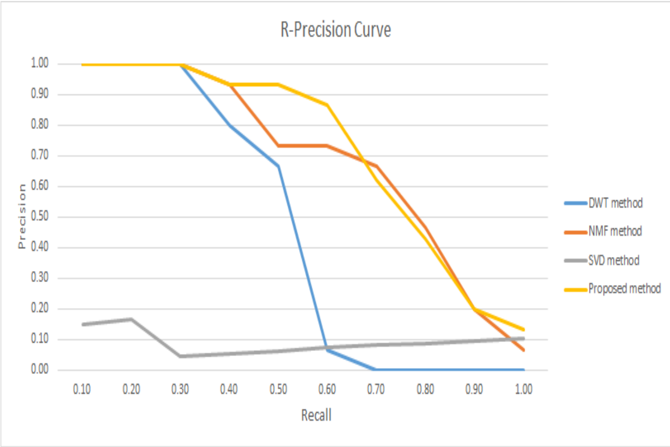


Figure 5.4: R-precision curve.

Table I. The Content-preserving operations and parameter values.

Tool	Manipulation	Parameter	Parameter values
PhotoELF	Brightness adjustment	PhotoELF scale	-20, -10, 10, 20
PhotoELF	Contrast adjustment	PhotoELF scale	-20, -10, 10, 20
PhotoELF	Gamma correction	PhotoELF scale	70, 90, 110, 130
Scriptjpg	JPEG compression	Quality factor	60, 70, 80, 90
MATLAB	Gaussian filtering	Standard deviation	0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9
MATLAB	Rotation and cropping	Angle in degree	1, 2, 3, 4, 5, 10, 15,, 30, 45, 90, -1, -2, -3, -4, -5, -10, -15, -30, -45, -90
MATLAB	Motion filtering	No. of pixels	9, 15
MATLAB	Salt and pepper noise	Noise density	0.05, 0.10, 0.15, 0.20
MATLAB	Circular average filtering	Radius	5, 10

Table II. Threshold values between original image and its distorted versions.

Wavelets	Attacks								
	brightness	circular filtering	Compression	contrast	gamma	Circular filtering	Median filtering	Rotated	Salt and pepper noise
Haar	0.08	0.08	0.08	0.08	0.06	0.08	0.08	0.08	0.08
Db4	0.07	0.08	0.08	0.08	0.05	0.08	0.08	0.07	0.07
Sym	0.07	0.08	0.08	0.08	0.05	0.08	0.08	0.08	0.08
Coif	0.07	0.08	0.08	0.08	0.05	0.08	0.08	0.07	0.07
Bior	0.08	0.09	0.09	0.09	0.06	0.09	0.09	0.08	0.08
Rbior	0.08	0.08	0.08	0.08	0.06	0.08	0.08	0.08	0.08

Table III. Threshold values of different images.

Wavelets	Haar	Db4	Sym	Coif	Bior	Rbior
Threshold	0.23	0.18	0.23	0.21	0.23	0.23

VI. REFERENCES

[1] Zhenjun Tang, Xianquan Zhang and Shichao Zhang "Robust Perceptual Image Hashing Based on Ring Partition and NMF." IEEE Trans. on Knowledge and Data Engineering, vol. 26, no. 3, pp. 711-724 March 2014.

[2] R. Venkatesan, S.-M. Koon, M.H. Jakubowski, and P. Moulin, "Robust Image Hashing," Proc. IEEE Int'l Conf. Image Processing, pp. 664-666, 2000.

[3] A. Swaminathan, Y. Mao, and M. Wu, "Robust and Secure Image Hashing," IEEE Trans. Information Forensics and Security, vol. 1, no. 2, pp. 215-230, June 2006.

[4] V. Monga and M.K. Mihcak, "Robust and Secure Image Hashing via Non-Negative Matrix Factorizations," IEEE Trans. Information Forensics and Security, vol. 2, no. 3, pp. 376-390, Sept. 2007.

[5] M. Schneider and S.F. Chang, "A Robust Content Based Digital Signature for Image Authentication," Proc. IEEE Int'l Conf. Image Processing, vol. 3, pp. 227-230, 1996.

[6] V. Monga and B.L. Evans, "Perceptual Image Hashing via Feature Points: Performance Evaluation and Tradeoffs," IEEE Trans. Image Processing, vol. 15, no. 11, pp. 3452-3465, Nov. 2006.

[7] C.Y. Lin and S.F. Chang, "A Robust Image Authentication System Distinguishing JPEG Compression from Malicious Manipulation," IEEE Trans. Circuits System and Video Technology, vol. 11, no. 2, pp. 153-168, Feb. 2001.

[8] D. Wu, X. Zhou, and X. Niu, "A Novel Image Hash Algorithm Resistant to Print-Scan," Signal Processing, vol. 89, no. 12, pp. 2415-2424, 2009.

[9] V. Monga, A. Banerjee, and B.L. Evans, "A Clustering Based Approach to Perceptual Image Hashing," IEEE Trans. Information Forensics and Security, vol. 1, no. 1, pp. 68-79, Mar. 2006.

[10] J. Fridrich and M. Goljan, "Robust Hash Functions for Digital Watermarking," Proc. IEEE Int'l Conf. Information Technology: Coding and Computing, pp. 178-183, 2000.

[11] F. Lefebvre, B. Macq, and J.-D. Legat, "RASH: Radon Soft Hash Algorithm," Proc. European Signal Processing Conf., pp. 299-302, 2002.

[12] J.S. Seo, J. Haitsma, T. Kalker, and C.D. Yoo, "A Robust Image Fingerprinting System Using the Radon Transform," Signal Processing: Image Comm., vol. 19, no. 4, pp. 325-339, 2004.

[13] C.Y. Lin and S.F. Chang, "A Robust Image Authentication System Distinguishing JPEG Compression from Malicious Manipulation," IEEE Trans. Circuits System and Video Technology, vol. 11, no. 2, pp. 153-168, Feb. 2001.

[14] S.S. Kozat, R. Venkatesan, and M.K. Mihcak, "Robust Perceptual Image Hashing via Matrix Invariants," Proc. IEEE Int'l Conf. Image Processing, pp. 3443-3446, 2004.

[15] Y. Lei, Y. Wang, and J. Huang, "Robust Image Hash in Radon Transform Domain for Authentication," Signal Processing: Image Comm., vol. 26, no. 6, pp. 280-288, 2011.

[16] C.D. Roover, C.D. Vleeschouwer, F. Lefebvre, and B. Macq, "Robust Video Hashing Based on Radial Projections of Key Frames," IEEE Trans. Signal Processing, vol. 53, no. 10, pp. 4020-4036, Oct. 2005.

[17] Y. Zhao, S. Wang, X. Zhang, and H. Yao, "Robust Hashing for Image Authentication Using Zernike Moments and Local Features," IEEE Trans. Information Forensics and Security, vol. 8, no. 1, pp. 55-63, Jan. 2013.

[18] Zhou B, Yao Y (2010) "Evaluating information retrieval system performance based on user preference." of Intelligent Information Systems 34(3):227–248.

[19] Sridhar, Digital Image Processing. Oxford University Press, India, 2011.

[20] B. Zhou and Y. Yao, "Evaluating information retrieval system performance based on user preference," Journal of Intelligent Information Systems, vol. 34, no. 3, pp. 227–248, 2010.

Govindaraj P. Completed Diploma in Electronics and Communication from Technical Board Karanataka and the B.E. degree in Electronics and Communication Engineering from Visvesvaraya Technological University in the year 2010 and 2013 respectively. Currently, he is pursuing the M.Tech. degree in Signal processing, Cambridge Institute of Technology (Affliated to Visvesaraya Technological University) Bengaluru, India. His current research interests include signal processing, perceptual image hashing and biomedical image processing.

Sandeep R. received the B.A. degree in Hindi from Mysore Hindi Prachar Parishad, the B.E. degree in Electronics and Communication Engineering and the M.Tech. Degree in Electronics Engineering from Visvesvaraya Technological University in the year 2001, 2006 and 2009 respectively. Currently, he is pursuing Ph.D. in the department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati, India and working as Associate Professor Cambridge Institute of Technology (Affliated to Visvesaraya Technological University) Bengaluru, India. He has both academic and industry experience. His current research interests include perceptual image hashing, perceptual video hashing, biometric hashing and biomedical image processing.