Computers and Electrical Engineering 91 (2021) 106931

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

# Copyright protection scheme for color images using extended visual cryptography☆

Sonal Kukreja *, Geeta Kasana, Singara Singh Kasana

*Computer Science and Engineering Department, Thapar Institute of Engineering and Technology, Patiala, Punjab, 147004, India*

ARTICLE INFO

*Keywords:*
Copyright protection
Curvelet transform
*k*-means clustering
Extended visual cryptography
Baker's Map

ABSTRACT

Most of the existing visual cryptography based copyright protection schemes create random looking shares which can create suspicion of some secret information being shared. In order to handle this issue, a copyright protection scheme based on curvelet transform and extended visual cryptography is proposed for color images. Robustness of the scheme against various attacks is achieved by using inter-layer low and middle frequency coefficients obtained by applying discrete curvelet transform on $R, G$, and $B$ components of host image. The curvelet inter-layer coefficients are used to create a master share. This master share and the watermark are used to create the ownership share by using the codebook. *XOR*-superimposition of master and ownership shares retrieve the watermark to prove the copyright. Security of the scheme is achieved by creating meaningful ownership shares; and by using Baker's Map for scrambling watermark and transformed host image. The scheme is robust to withstand several image processing attacks as well as provides better imperceptibility. Effectiveness of the scheme is shown by comparing it with the existing copyright protection schemes.

## 1. Introduction

With the rapid advancements in information and multimedia technologies, security of the confidential data being transmitted on the network has become an important concern. The digital data, being transmitted can be in any form like text, images, audio, video *etc.* Transmitted data need to be protected from unauthorized users. In last few years, many schemes have been proposed to enforce security of digital data. Secure Digital Imaging is a crucial research area which includes schemes of cryptography, watermarking and steganography. Most commonly used types of schemes to secure the data are: cryptography and watermarking. In cryptography, the data is secured by scrambling it into unreadable form with the help of some key(s), and the same/different key(s) is used to unscramble it at the receiver side. Various properties of data security like data integrity, data confidentiality, authentication, non repudiation of data *etc*. are fulfilled by cryptography.

Watermarking is a scheme of inserting secret data into the original cover host image, using different approaches of embedding. It has been used in modern communication technologies, as a means to resolve the dispute and prove the copyright of any media. The images can be protected by inserting some extra data into them which would help in their protection. The embedding of data in the cover image should result into minimum distortion to ensure imperceptibility and the marked image should be robust to various image processing attacks.

Visual Cryptography (*VC*) or Visual Secret Sharing (*VSS*) proposed by Naor and Shamir [1], is a variation of cryptography to share the secret image data. This scheme creates a number of random looking shares for the secret image and then these shares

☆ This paper was recommended for publication by associate editor E. Cabal-Yepez.

* Corresponding author.
*E-mail addresses:* kukreja.sonal@gmail.com (S. Kukreja), gkasana@thapar.edu (G. Kasana), singara@thapar.edu (S.S. Kasana).

https://doi.org/10.1016/j.compeleceng.2020.106931
Received 26 December 2019; Received in revised form 3 September 2020; Accepted 19 November 2020
Available online 13 February 2021
0045-7906/

are distributed among a number of participants. These shares do not have abilities to disclose any information about the secret image until they are superimposed in a definite prescribed manner. Also minimum number of qualified shares can participate in the superimposition to recover the secret image. The advantage of this approach is that no complex computation is required to retrieve the secret image back, as it is obtained by just overlapping the shares. This scheme has further been improved to work for color images [2].

*VSS* can be combined with watermarking in two ways. In the first way, the watermark is processed to generate one or more shares by using *VSS*, and these shares are then embedded or combined with the host image to generate a watermarked image. The other way is, that one share image is generated from the host image, which is then combined with the watermark to generate another share which is referred as the key or Ownership share and is stored with the Certified Authority (*CA*). This share is later used to prove the copyright of the image in the case of disputes. The main advantage of using *VSS* with watermarking is that no complex computations or computerized extracting algorithms are required to extract the watermark, as it can be extracted simply by superimposing, the results of which are directly visible to human eyes.

The idea of *VSS* based watermarking was initiated by Hou and Chen [3]. They generated two random looking shares from the watermark using *(2,2)-VSS*, with one share embedded into the host image while another share is stored with the *CA* as a secret key share. To retrieve the watermark for claiming copyright, the secret key share is superimposed with the share extracted from received marked image. The scheme proved to be robust but required an additional data hiding process to embed the share which does not utilize the properties of *VSS* efficiently. Hwang et al. [4] modified Hou and Chen scheme [3] by generating two shares from the host image that are known as the Master Share (*MS*) and Ownership Share (*OS*). *MS* is constructed solely from the original image while *OS* is constructed using *MS* and Watermark and is registered with *CA*. *MS* is constructed by randomly selecting pixels of the original image using a secret key. Then *OS* is constructed by using the most significant bit of every selected pixel and a binary watermark. *OS* is later superimposed with the *MS* generated from received image to retrieve the watermark, which is further used to verify the copyright of the received image. This scheme proved to be robust against various attacks but does not work effectively for all the gray images, especially the images that have left-skewed or right-skewed histograms [5]. To address this issue, Hsu and Hou [6] proposed a scheme in spatial domain where *OS* is constructed using binary secret message bit, global mean intensity of the image and mean of the neighboring pixel values of randomly selected pixels in the image. The scheme proved to be robust against many attacks except cropping attack.

Wang and Chen [7] proposed a hybrid *DWT–SVD* based copyright protection method using *VSS*. The features are extracted from the host image using *DWT* and *SVD* transforms. These extracted features are then classified into two classes using k-means clustering technique to construct the *MS*. *OS* is constructed using the same strategy as the existing schemes. Their scheme outperformed existing schemes in robustness against most of the image processing attacks but the restriction on the watermark size prevailed. Robustness and security of the host image were further improved by Rawat et al. [8] by using Fractional Fourier Transform (*FrFT*) to create the *MS*. *FrFT* ensures high security as the watermark can be extracted only if the orders of the *FrFT* are known to the attacker. The scheme still showed low weakness to cropping, impulse noise, Gaussian noise and sharpening attacks.

Some other copyright protection schemes [9–11] were proposed using *VC* and other features like *DWT , SVD*, *etc*., which were robust to many attacks. These schemes create random looking shares and the size of watermark is fixed. Shao et al. [12] proposed watermarking scheme for color images based on quaternion-type moment invariants and *VC*, which helps in enhancing robustness but created meaningless shares. Mingfu et al. [13] proposed an image authentication scheme using Scale Invariant Feature Transform (*SIFT*) keypoints where a hash key is generated using local binary features and *SIFT* keypoints. This hash key is embedded into the host image which is later used at the receiver's side to prove the authentication. Ali et al. [14] proposed an image watermarking scheme that uses canny edge detection and support vector machine to create the master share. This scheme improved the robustness against various attacks except rotation and also suffered from pixel expansion.

The main limitation of these mentioned schemes is that they generate meaningless shares, that create a suspicion to the third party attackers that some secret information is being stored or shared. In most of these schemes, size of the watermark is also restricted and low robustness is observed against certain attacks.

In the proposed scheme, the use of non fine scale layer curvelet coefficients has helped in improving the robustness against various attacks. There is no restriction on watermark size. Baker's map has been used to scramble the watermark and host image to enhance security. A codebook is also proposed that helps in creating meaningful ownership shares, thereby enhancing security of the scheme. The process of extracting the watermark uses mechanism of visual cryptography *i.e.* the watermark emerges naturally by superimposing shares using *XOR* operation, without any complex arithmetic computation. Thus the scheme provides blind extraction.

The paper has been organized into following Sections: Section 2 discusses the background of the proposed scheme like *DCuT* and Baker's map. The proposed scheme has been described in Section 3. Experimental Results and Performance Analysis have been illustrated in Section 4 followed by the conclusion and future scope in Section 5.

## 2. Preliminaries

In this section, *DCuT* and Baker's Map, used in the proposed scheme, are discussed.

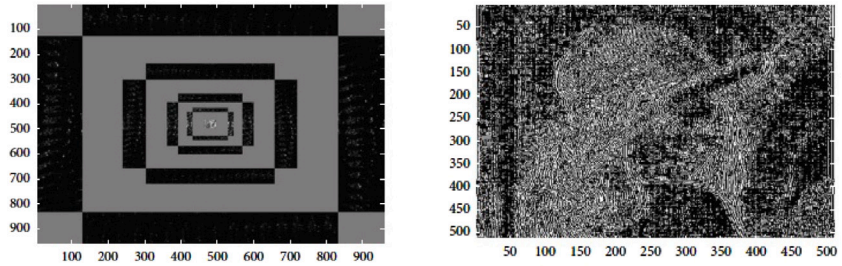



**Fig. 1.** The coefficient matrix image of (a) first five layers (b) sixth layer *via* Curvelet transform for Lena.

### 2.1. Discrete curvelet transform

*DCuT*, proposed by Candes and Donoho [15] is a multiresolution transform that provides a sparser representation of images in comparison to other wavelet transforms like *FrFT* and Discrete Wavelet Transform *(DWT)*. *FrFT* and *DWT* require a larger number of frequency coefficients and wavelets basis functions, respectively to represent the image. *DCuT* has an efficient capability to represent edges and curves information of the image. Taking a two dimensional image of the form $f(x, y)$ as an input, *DCuT* generates Curvelet Coefficients using:

$$C^D(j,l,k) = \sum_{0 \le x,y \le n} f[x,y]^{\theta^D_{j,l,k}(x,y)} \tag{1}$$

Here, $\theta^D_{j,l,k}(x,y)$ is a digital curvelet waveform of a cartesian array of form $f[x,y]$ where $0 \le x, y \le n$. $n \times n$ represents size of the image. $j$, $l$ and $k$ are scale, orientation and translation parameters respectively. The authors presented two variations for *DCuT*: Unequi-spaced Fast Fourier Transform *FFT* and Frequency wrapping (*WRAP*). *WRAP* is used in most of the image watermarking and compression applications as it has easier implementation and less computation time compared to Unequi-spaced *FFT*.

The coefficients obtained from curvelet transform are of structure $C(j,l,k)$, where $j$, $l$ and $k$ represent scale, direction and translation parameters, respectively. The curvelet transform coefficients are divided into three frequency sub-bands: Low Frequency *(LF)*, Middle Frequency *(MF)* and High Frequency *(HF)*. The number of cells in these sub-bands depend on the image size. For example, images of size 512 × 512 are decomposed using *WRAP* based *DCuT* into different cells, such as C(1,1) to C(1,6), with 6 scale parameters and 16 orientation parameters. The innermost cell C(1,1) contains *LF* coefficients, middle cells C(1,2) to C(1,5) contain *MF* coefficients and the outermost C(1,6) contains *HF* coefficients and is equal to the size of the image. The reason behind using *LF* and *MF* is that they provide better robustness to the scheme. Even though when the image undergoes any attack, the coefficients help in retrieving the watermark and prove the copyright. Fig. 1 represents coefficient matrix images of the first five layers and outermost layer for Lena image, constructed using *DCuT*.

### 2.2. Baker's map

A Baker's map [16] is a two-dimensional chaotic bijection of a unit square matrix onto itself obtained by shuffling the pixels positions of the image without affecting their values. The transformation can be achieved in two ways [17], either by folding over one of the sliced halves onto other or the upper section remains unfolded. These cases have been represented mathematically in (2), as follows:

$$I(x,y) = \begin{cases} [x_{i+1}, y_{i+1}] = [2x_i, \frac{y_i}{2}], & \text{for } 0 \le x_i \le \frac{1}{2} \\ [x_{i+1}, y_{i+1}] = [2(1-x_i), 1-\frac{y_i}{2}], & \text{for } \frac{1}{2} \le x_i \le 1 \end{cases} \tag{2}$$

where $x_0$ and $y_0$ are initialized with some random values .

$$BM(x,y) = \begin{cases} 0, & \text{for } 0 \le x_i \le \frac{1}{2} \\ 1, & \text{for } \frac{1}{2} \le x_i \le 1 \end{cases} \tag{3}$$

The image is scrambled by applying *XOR* operation between the instances of Baker binary matrix obtained through (3) and the corresponding bit in the unscrambled image. To retrieve the unscrambled image, these steps are applied in reverse order.

Analysis of different scrambling methods [18] shows that Baker's Map provides better scrambling results as compared to Arnold and Henon Map Transform. *PSNR* of scrambled image by Baker's Map is lesser than by Arnold and Henon Map Transforms. Also, Baker's Map is much faster as compared to the other scrambling transforms.

## 3. Proposed scheme

The proposed scheme is divided into two phases: Share Construction and Copyright Verification. Implementation of these two phases are shown in Figs. 2 and 3.

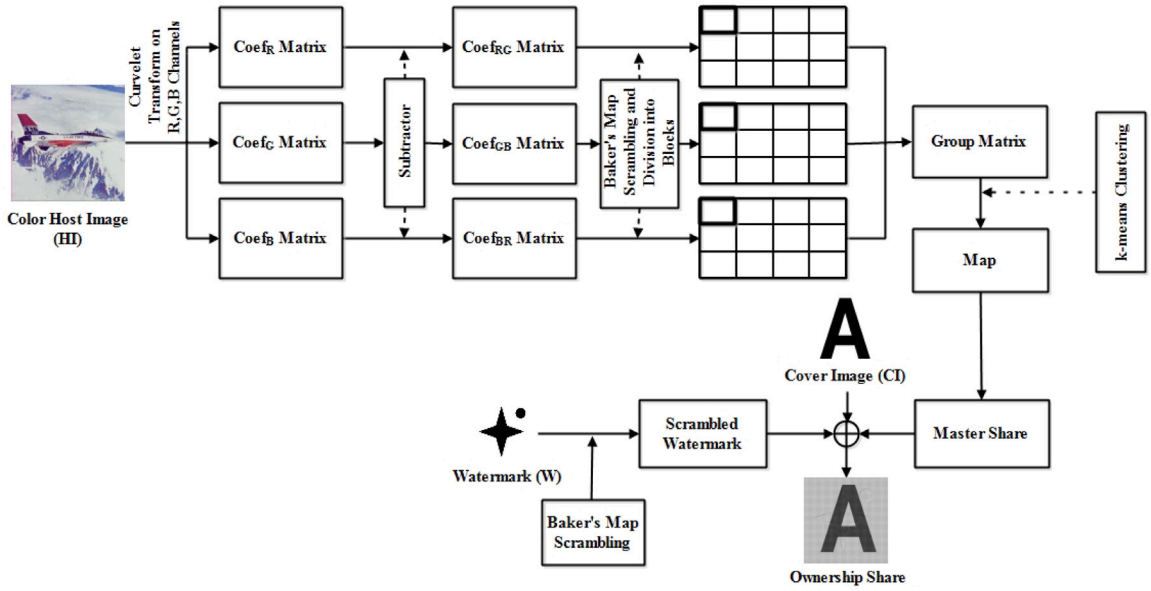



**Fig. 2.** Block Diagram for Share Construction Phase.

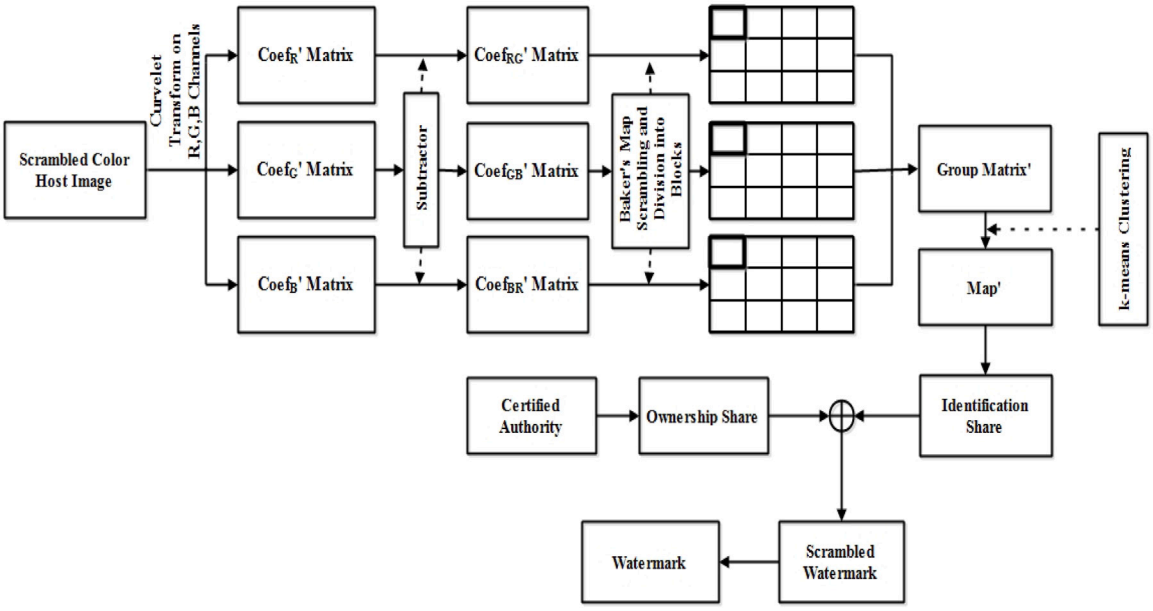



**Fig. 3.** Block Diagram for Copyright Verification Phase.

### *3.1. Share construction*

In this phase, all channels of the color image are used to generate the binary *MS*. *DCuT via* Wrapping is applied individually on all channels of the image. As human eyes are more sensitive to *LF* information than *HF*, the inner non-fine scale layers that include *LF* and *MF* sub-bands, are used to construct the coefficient matrix for every channel, that helps in enhancing robustness of the scheme. The constructed coefficient matrices are further processed through subtractors. The main motivation behind the usage of subtractors is that every transformed component now contains information about two layers and gives better robustness performance. These matrices are further combined to create a single matrix. To enhance the security of the proposed scheme, Baker's map is applied to both the obtained coefficient matrix and watermark.

*MS* is constructed by applying *k*-means clustering on the obtained coefficient matrix. *MS*, Watermark ($W$) and Cover Image ($CI$) are then used to construct a meaningful *OS* using a codebook that has been prepared considering the following criteria:

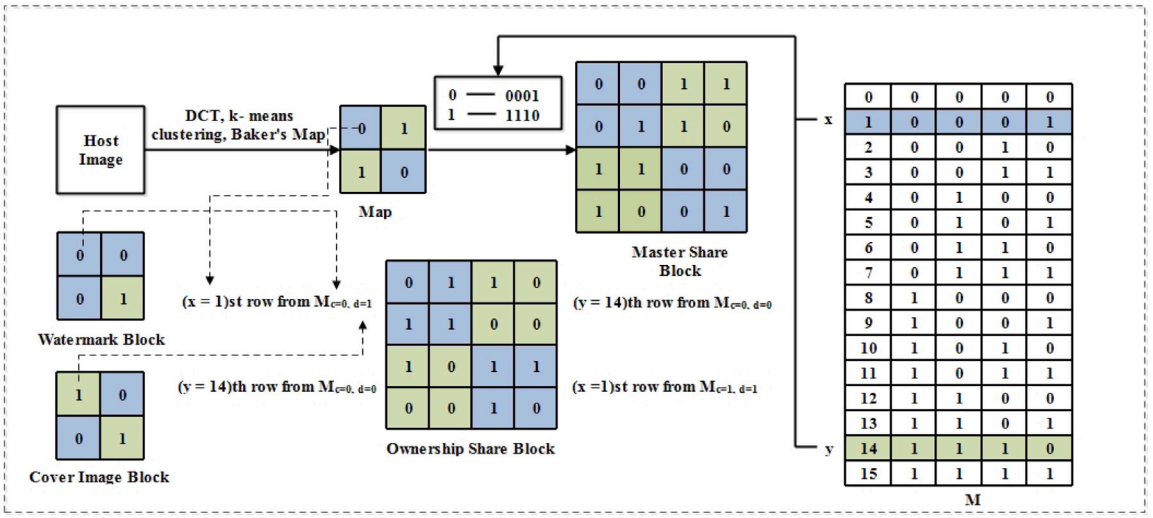



**Fig. 4.** An Example to show (a) Construction of Ownership Share and (b) Superimposition of Identification Share and Ownership Share.

1. The pixels in the *OS* block should resemble the corresponding bit in *CI*, *i.e.* if $CI_{bit} == 0$, count of 0 > count of 1 in corresponding *OS* block and if $CI_{bit} == 1$, count of 1 > count of 0 in corresponding *OS* block.
2. *XOR*-superimposition of corresponding *OS* and *MS* block bits should result in a *W* block, that can be reduced to the watermark bit, *i.e.* if $W_{bit} == 0$, the pixels in corresponding *OS* block should be as much similar as possible to the *MS* block and if $W_{bit} == 1$, the pixels in corresponding *OS* block should be as different as possible to the *MS* block.

Four matrices are represented as $M_{c,d}$ where $c$ and $d$ represent bit values of $W$ and $CI$, which are in set $\{0, 1\}$.

$$M_{00} = \begin{bmatrix} 0&0&0&0\\ 0&0&0&1\\ 0&0&1&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&1&0&0\\ 0&1&0&0\\ 0&1&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0 \end{bmatrix}, \quad M_{01} = \begin{bmatrix} 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 1&1&1&0\\ 1&1&0&1\\ 1&1&1&0\\ 1&0&1&1\\ 1&1&1&0\\ 1&1&0&1\\ 1&1&1&0\\ 1&1&1&1 \end{bmatrix}, \quad M_{10} = \begin{bmatrix} 0&0&0&1\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 1&0&0&0\\ 0&0&0&1\\ 0&0&0&1\\ 0&0&0&1\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0\\ 0&0&0&1\\ 0&0&0&1 \end{bmatrix}, \quad M_{11} = \begin{bmatrix} 1&1&1&0\\ 1&1&1&0\\ 1&1&0&1\\ 1&1&1&0\\ 1&0&1&1\\ 1&0&1&1\\ 1&0&1&1\\ 1&1&1&0\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1\\ 0&1&1&1 \end{bmatrix} \tag{4}$$

After the construction of $OS$, $W$ is kept secretly by the copyright owner and the $OS$ is registered to a Certified Authority ($CA$) and used in the copyright verification, when required.

The steps for this phase are described in Algorithm 1 and an example has been shown in Fig. 4(a) to construct an $OS$ block from $MS$ and $W$ blocks. In Fig. 4(a), blocks of size $2\times2$ have been taken from $CI$ and $W$. The $Map$ shown is created from the $HI$ using the steps described in Algorithm 1. Two random rows, $x = 1$ and $y = 14$ have been chosen from $M$ to represent 0 and 1, respectively. Hence, $MS$ is constructed by replacing 0 and 1 in $Map$ with patterns "0001" and "1110", respectively. As the first bit in $Map$ is 0, the $x^{th}$ row is chosen from $M_{c=0,d=1}$ and placed in corresponding OS block, where $c$ and $d$ represent corresponding bits in $W$ and $CI$ respectively.

### 3.2. Copyright verification

In this phase, an Identification Share ($IS$) is generated from the received host image using the similar Master Share construction steps followed in previous phase. The $OS$ registered with $CA$ is retrieved and superimposed with $IS$ to retrieve an expanded watermark. The watermark is resized to its original size and compared with the original watermark stored with owner to prove the copyright. Steps for this phase are described in Algorithm 2 and an example is shown in Fig. 4-(b) to show the superimposition of $IS$ and $OS$ blocks to retrieve the watermark block.

## 4. Experimental results and analysis

The performance of the proposed scheme is investigated through a set of experiments performed using MATLAB (R2018a), 64-bit (win64) software, on different color images of size 512 × 512 and binary watermark image of size 256 × 256. Size of watermark is dynamic as it can be increased or decreased as per host image size. Number of blocks created for the host image should be equal to the size of watermark. Ten host images *viz.* Airplane, Barbara, Girl, Goldhill, House, Lake, Lena, Mandrill, Peppers and Zelda are used for experimentation and shown in Fig. 5. The cover and watermark images used in the experimentation are shown in Fig. 6.

The implementation results of the proposed scheme are shown in Fig. 7 for Mandrill image. It can be seen in Fig. 7(c) that a meaningful ownership share is created. When master share (Fig. 7(b)) and ownership share (Fig. 7(c)) are superimposed with $XOR$-operation, watermark shown in Fig. 7(d) is extracted, which is further processed to get its original size, as shown in Fig. 7(e).

### 4.1. Computational complexity of proposed scheme

The computational complexity of Algorithm 1 is $O(n^2 log\ n)$. This can be explained as follows: Complexity for step 1 is $O(n^2 log\ n)$ [15]. Steps 2–5 take $O(n^2)$ time each, as every pixel in $n \times n$ image is used in these pre-processing steps while steps 6–8 take $O(\frac{n^2}{bh \times bw})$ time each. This can be represented as,

$$Complexity = O(n^2 log\ n) + O(4 \times (n^2)) + O(3 \times (\frac{n^2}{bh \times bw})) \tag{10}$$

On the same lines, one can find that computational complexity of Algorithm 2 is $O(n^2 log\ n)$ as its step 1 takes $O(n^2 log\ n)$ [15], steps 2–6 take $O(n^2)$ time as every pixel in $n \times n$ image is used to create $MS$ while step 8 takes $O(n^2)$ time as $XOR$-operation is performed for all pixels of the shares.

### 4.2. Robustness assessment

Parameters used to measure robustness of the scheme are Normalized Correlation ($NC$), Structural Similarity Index Measure ($SSIM$) and Bit Error Rate ($BER$). These are described as follows:

**Algorithm 1** Share Construction Phase

**Input:** Host Image ($HI$) of size $hh \times hw$, Watermark $W$ of size $wh \times ww$, Cover Image ($CI$) of size $ch \times cw$
**Output:** Master Share ($MS$) and Ownership Share ($OS$)

1: Apply Curvelet decomposition on *R,G,* and *B* channels to get their coefficient matrices. Extract the *LF* and *MF* sub-bands from these matrices and combine theme to store as $Coef_R, Coef_G, Coef_B$, respectively for every channel. These sub-bands are further processed through subtractors to obtained inter-layer transformed component matrices ($Coef_{RG}, Coef_{GB}, Coef_{BR}$), described in equations 4 to 6.

$$Coef_{RG} = Coef_R - Coef_G \tag{5}$$

$$Coef_{GB} = Coef_G - Coef_B \tag{6}$$

$$Coef_{BR} = Coef_B - Coef_R \tag{7}$$

2: Scramble $Coef_{RG}$, $Coef_{GB}$, $Coef_{BR}$ and $W$ using Baker's Map.
3: Segment the coefficient matrices into blocks $b_i$ of size $bh \times bw$, where $i = 1$ to $n$, $n = \lceil \frac{hh \times hw}{bh \times bw} \rceil$. The matrices of blocks are represented as $rblock_i$, $gblock_i$, $bblock_i$
4: **for** i= 1 to $n$ **do**
Concatenate the corresponding blocks of three matrices into a single matrix $C_i$
5: Classify the blocks in $C_i$ into 2 clusters: Cluster 0 and 1, using *k*-means clustering and store them as *Map* that contains cluster number for every corresponding block.
6: Construct *MS* corresponding to *Map* using the Matrix $M$ given in (8). Choose 2 different rows from the matrix $M$ for *Map* values '0' and '1'. Let the decimal value of the chosen row for '0' and '1' are represented by $x$ and $y$ respectively. Replace '0' and '1' in the *Map* with the chosen $x^{th}$ and $y^{th}$ rows, respectively to construct *MS*.

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{8}$$

7: Construct *OS* using *Map*, $W$ and $CI$ by following ((4)). This procedure has been explained as follows:
8: **for** i: 1 to mw **do**
9: **for** j: 1 to mh **do**
10: **if** *Map*(i,j)==0 **then**
Choose $x^{th}$ row from $M_{c,d}$ and place it at the corresponding position in *OS*.
11: **else**
Choose $y^{th}$ row from $M_{c,d}$ and place it at the corresponding position in *OS*.


**Fig. 5.** Different host images (a) Airplane (b) Barbara (c) Girl (d) Goldhill (e) House (f) Lake (g) Lena (h) Mandrill (i) Peppers (j) Zelda.
*Source:* (Image source: http://sipi.usc.edu/database/database.php?volume = misc).

**Algorithm 2** Copyright Verification

**Input:** Attacked Host Image (*HI'*) of size $hh \times hw$

**Output:** Identification Share (*IS*)

1: Apply Curvelet decomposition on *R,G,* and *B* channels to get their coefficient matrices. Extract the *LF* and *MF* sub-bands from these matrices and combine theme to store as $Coef_R, Coef_G, Coef_B$, respectively for every channel. These sub-bands are further processed through subtractors to obtained inter-layer transformed component matrices ($Coef_{RG}, Coef_{GB}, Coef_{BR}$), described in equations 3 to 5.

2: Scramble $Coef_{RG}$, $Coef_{GB}$, $Coef_{BR}$ and *W* using Baker's Map.

3: Segment the coefficient matrices into blocks $b_i$ of size $bh \times bw$, where $i = 1$ to $n$, $n = \lceil \frac{hh \times hw}{bh \times bw} \rceil$. The matrices of blocks are represented as $rblock'_i$, $gblock'_i$, $bblock'_i$

4: **for** i= 1 to $n$ **do**

Concatenate the corresponding blocks of three matrices into a single matrix $C_i$

5: Classify the blocks in $C_i$ into 2 clusters: Cluster 0 and 1, using *k*-means clustering and store them as *Map'* that contains cluster number for every corresponding block.

6: Construct *IS* corresponding to *Map'* using the Matrix *M* given in (8). Choose the same 2 rows from the matrix *M* for *Map* values '0' and '1', that were chosen during the Share Construction Phase. Let the decimal value of the chosen row for '0' and '1' are represented by $x$ and $y$ respectively. Replace '0' and '1' in the *Map* with the chosen $x^{th}$ and $y^{th}$ row, respectively to construct *IS*.

7: Retrieve *OS* from *CA*.

8: Retrieve Watermark Image *W'* by stacking *OS* and *IS* using *XOR* operation.

9: Perform reduction process to obtain reduced *RW'* of size $wh \times ww$ by following rules:

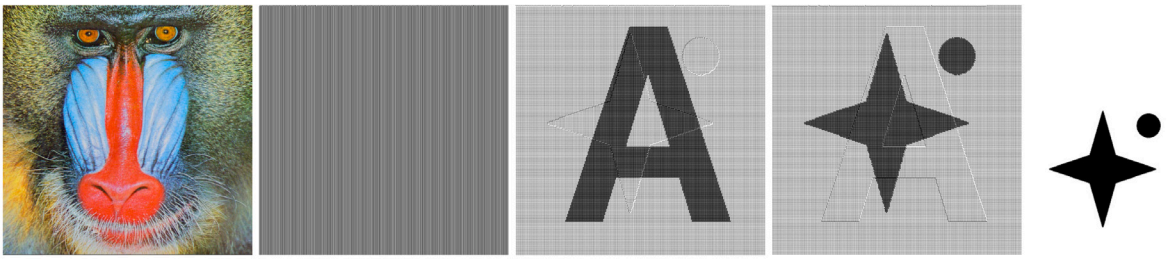
$$w = \begin{cases} 1, & if \quad \sum_j \sum_k s'_{j,k} \geq 2 \\ 0, & if \quad \sum_j \sum_k s'_{j,k} < 2 \end{cases} \tag{9}$$

where $w$ is a binary pixel in *RW'*, $s'_{j,k}$ represent pixels in *W'* blocks of size $2 \times 2$.

10: Scramble the Watermark *W'* to obtain descrambled watermark *W''*.

(a) (b)

**Fig. 6.** (a) Cover Image (b) Watermark.



(a) (b) (c) (d) (e)

**Fig. 7.** (a) Host Image (b) Master Share (c) Ownership Share (d) Extracted Watermark (e) Reduced Watermark.

#### 4.2.1. Normalized correlation

*NC* is used to measure the correlation between the original watermark and the extracted watermark and is defined as:

$$NC = \frac{\sum_{m=1}^{N_s} \sum_{n=1}^{N_s} \overline{(W(m,n) \oplus W'(m,n))}}{N_s \times N_s} \tag{11}$$

where $W(m,n)$ and $W'(m,n)$ represent the original watermark and the extracted watermark respectively, $\oplus$ denotes the exclusive-or (*XOR*) operation and $N_s \times N_s$ is the size of the watermark image.

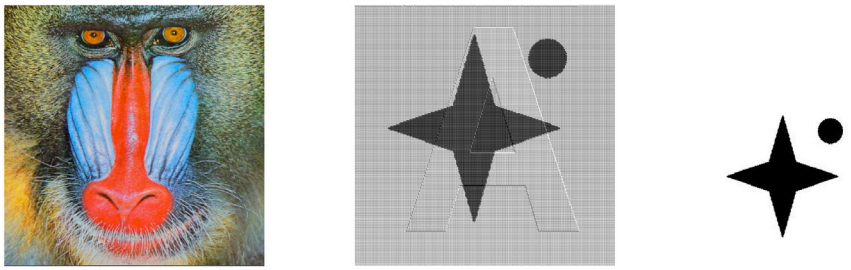



**Fig. 8.** (a) Compressed Image ($Q$=90, $PSNR$ = 33.45) (b) Superimposed Result (c) Reduced Watermark ($NC$ = 1.00).

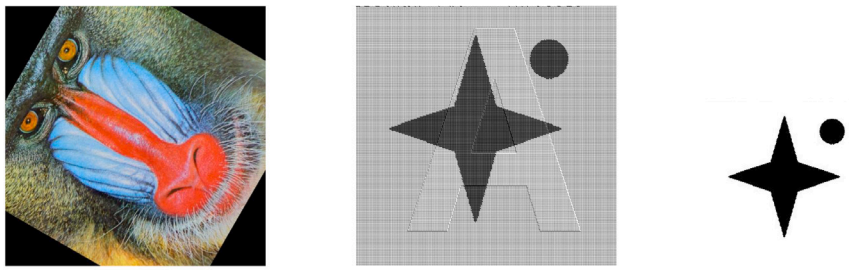



**Fig. 9.** (a) Rotated Image ($A$=60°, $PSNR$ = 9.4488) (b) Superimposed Result (c) Reduced Watermark ($NC$ = 0.9993).

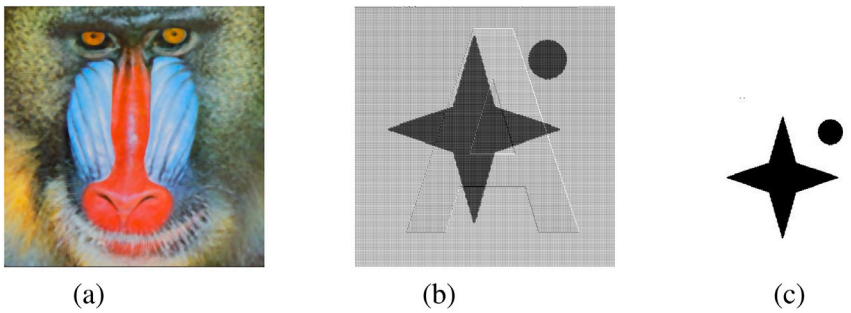



**Fig. 10.** (a) Image after Median Filtering Attack ($ws$ = 3 × 3, $PSNR$ = 20.4138) (b) Superimposed Result (c) Reduced Watermark ($NC$ = 1.00).

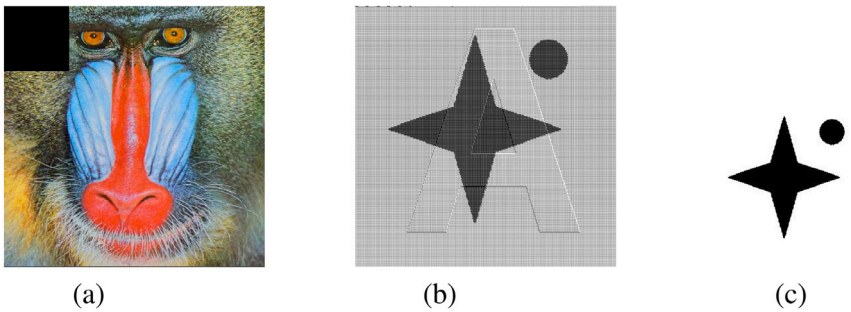



**Fig. 11.** (a) Cropped Image ($cw$ = 128 × 128, $PSNR$ = 18.0368) (b) Superimposed Result (c) Reduced Watermark ($NC$ = 0.9997).

**Table 1**
*NC* for Robustness test on different images .

| Images | Airplane | | | Barbara | | | Girl | | | Goldhill | | | House | | | Lake | | | Lena | | | Mandrill | | | Peppers | | | Zelda | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER |
| JPEG Compression | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Q=40 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9998 | 1 | 0.00016 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Q=50 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9998 | 1 | 0.00016 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Q=60 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9998 | 1 | 0.00016 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Q=70 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9998 | 1 | 0.00016 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Q=80 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9998 | 1 | 0.00016 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Q=90 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9998 | 1 | 0.00016 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Rotation | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| A=10 | 0.9998 | 1 | 0.00018 | 0.9996 | 1 | 0.00038 | 0.9998 | 1 | 0.00018 | 0.9996 | 1 | 0.00044 | 0.9997 | 1 | 0.00028 | 0.9997 | 1 | 0.0003 | 0.9997 | 1 | 0.00033 | 0.9997 | 1 | 0.0003 | 0.9997 | 1 | 0.00025 | 0.9993 | 1 | 0.00065 |
| A=20 | 0.9996 | 1 | 0.00038 | 0.9994 | 1 | 0.00056 | 0.9996 | 1 | 0.00036 | 0.9993 | 1 | 0.00074 | 0.9995 | 1 | 0.00047 | 0.9995 | 1 | 0.0005 | 0.9996 | 1 | 0.00044 | 0.9995 | 1 | 0.00054 | 0.9995 | 1 | 0.00048 | 0.9991 | 1 | 0.00085 |
| A=30 | 0.9995 | 1 | 0.00051 | 0.9994 | 1 | 0.00059 | 0.9995 | 1 | 0.0005 | 0.9991 | 0.9999 | 0.00094 | 0.9994 | 1 | 0.00057 | 0.9994 | 1 | 0.00061 | 0.9995 | 1 | 0.00053 | 0.9993 | 1 | 0.00065 | 0.9994 | 1 | 0.00064 | 0.9991 | 1 | 0.00093 |
| A=40 | 0.9995 | 1 | 0.00054 | 0.9993 | 1 | 0.00067 | 0.9995 | 1 | 0.00053 | 0.9989 | 0.9999 | 0.0011 | 0.9993 | 1 | 0.00068 | 0.9994 | 1 | 0.00064 | 0.9995 | 1 | 0.00051 | 0.9993 | 1 | 0.0007 | 0.9994 | 1 | 0.00059 | 0.9991 | 1 | 0.00091 |
| A=50 | 0.9994 | 1 | 0.00056 | 0.9993 | 1 | 0.00068 | 0.9995 | 1 | 0.00054 | 0.9988 | 0.9999 | 0.0012 | 0.9993 | 1 | 0.0007 | 0.9993 | 1 | 0.00065 | 0.9994 | 1 | 0.00056 | 0.9993 | 1 | 0.00073 | 0.9994 | 1 | 0.00062 | 0.9991 | 1 | 0.00093 |
| A=60 | 0.9994 | 1 | 0.00057 | 0.9993 | 1 | 0.0007 | 0.9994 | 1 | 0.00056 | 0.9988 | 0.9999 | 0.0012 | 0.9993 | 1 | 0.00071 | 0.9993 | 1 | 0.0007 | 0.9994 | 1 | 0.00059 | 0.9993 | 1 | 0.00073 | 0.9993 | 1 | 0.00065 | 0.999 | 1 | 0.00099 |
| Median Filter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| WS=2*2 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| WS=3*3 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| WS=4*4 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| WS=5*5 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.00003 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| WS=6*6 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.00003 | 1 | 1 | 0 | 1 | 1 | 0.000045 |
| WS=7*7 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.00003 | 1 | 1 | 0.000015 | 1 | 1 | 0.00003 |
| Cropping | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| cw=16*16 | 1 | 1 | 0.000001 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.00009 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| cw=32*32 | 1 | 1 | 0.000015 | 1 | 1 | 0.000045 | 1 | 1 | 0 | 1 | 1 | 0.00003 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.0001 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000045 |
| cw=64*64 | 0.9999 | 1 | 0.00006 | 0.9999 | 1 | 0.000061 | 1 | 1 | 0.00003 | 1 | 1 | 0.000045 | 1 | 1 | 0.000045 | 0.9998 | 1 | 0.00015 | 0.9999 | 1 | 0.000061 | 0.9999 | 1 | 0.000076 | 1 | 1 | 0.000045 | 0.9999 | 1 | 0.00009 |
| cw=128*128 | 0.9998 | 1 | 0.00018 | 0.9998 | 1 | 0.00021 | 0.9998 | 1 | 0.00019 | 0.9997 | 1 | 0.00025 | 0.9998 | 1 | 0.00018 | 0.9997 | 1 | 0.00032 | 0.9998 | 1 | 0.00022 | 0.9997 | 1 | 0.00025 | 0.9998 | 1 | 0.00019 | 0.9997 | 1 | 0.0003 |
| cw=256*256 | 0.9993 | 1 | 0.00074 | 0.9992 | 1 | 0.00077 | 0.9992 | 1 | 0.00077 | 0.9991 | 0.9999 | 0.00093 | 0.9992 | 0.9999 | 0.0008 | 0.9991 | 0.9999 | 0.00086 | 0.9991 | 0.9999 | 0.0008 | 0.9992 | 0.9999 | 0.00082 | 0.9992 | 1 | 0.00077 | 0.999 | 0.9999 | 0.00099 |
| Gaussian Noise | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| V=0.01 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.0001 |
| V=0.03 | 1 | 1 | 0.000015 | 1 | 1 | 0.000045 | 1 | 1 | 0 | 1 | 1 | 0.000045 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.00003 | 1 | 1 | 0 | 0.9999 | 1 | 0.00013 |
| V=0.05 | 1 | 1 | 0.000015 | 1 | 1 | 0.000045 | 1 | 1 | 0 | 0.9998 | 1 | 0.00018 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.00003 | 1 | 1 | 0 | 0.9997 | 1 | 0.00025 |
| V=0.07 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000076 | 1 | 1 | 0 | 0.9998 | 1 | 0.00019 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.00003 | 0.9999 | 1 | 0.000076 | 1 | 1 | 0 | 0.9997 | 1 | 0.00028 |
| V=0.09 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000076 | 1 | 1 | 0.000015 | 0.9997 | 1 | 0.00027 | 1 | 1 | 0.00003 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000045 | 0.9999 | 1 | 0.000076 | 1 | 1 | 0.000015 | 0.9997 | 1 | 0.0003 |
| Poisson Noise | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.00009 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| Salt and Pepper Noise | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| D=0.01 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| D=0.03 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| D=0.05 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000045 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.00003 | 1 | 1 | 0 | 0.9999 | 1 | 0.0001 |
| D=0.07 | 1 | 1 | 0.000015 | 1 | 1 | 0.000045 | 1 | 1 | 0 | 0.9999 | 1 | 0.0001 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.00003 | 1 | 1 | 0 | 0.9998 | 1 | 0.00016 |
| D=0.09 | 1 | 1 | 0.000015 | 1 | 1 | 0.000045 | 1 | 1 | 0 | 0.9999 | 1 | 0.00012 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.00003 | 1 | 1 | 0.000045 | 1 | 1 | 0.000015 | 0.9998 | 1 | 0.00016 |
| Speckle Noise | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| V=0.01 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| V=0.03 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| V=0.05 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| V=0.07 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| V=0.09 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| Sharpening | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| R=2, A=0.2 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| R=2, A=0.5 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 |

(*continued on next page*)

**Table 1** (*continued*).

| Images | Airplane | | | Barbara | | | Girl | | | Goldhill | | | House | | | Lake | | | Lena | | | Mandrill | | | Peppers | | | Zelda | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER | NC | SSIM | BER |
| R=2, A=0.8 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| R=2, A=1.1 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| R=1, A=1.4 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| R=1, A=1.7 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.00003 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| R=1, A=2.0 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.00003 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| Sobel | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0.9983 | 0.9999 | 0.0017 | 0.9986 | 0.9999 | 0.0014 | 0.9977 | 0.9998 | 0.0023 | 0.9997 | 1 | 0.0003 | 0.9988 | 0.9999 | 0.0012 | 0.9989 | 0.9999 | 0.0011 | 0.9983 | 0.9999 | 0.0017 | 0.9988 | 0.9999 | 0.0012 | 0.9991 | 0.9999 | 0.0008 | 0.9984 | 0.9999 | 0.0016 |
| Blurring | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| R=5 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0.00003 |
| R=10 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| R=15 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 1 | 1 | 0 | 1 | 1 | 0.000015 |
| R=20 | 1 | 1 | 0.000015 | 1 | 1 | 0.00006 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.00003 | 1 | 1 | 0.000045 | 1 | 1 | 0.000015 | 0.9999 | 1 | 0.000076 |
| R=25 | 1 | 1 | 0.000015 | 1 | 1 | 0.000091 | 1 | 1 | 0 | 1 | 1 | 0.000015 | 1 | 1 | 0.00003 | 0.9999 | 1 | 0.000091 | 1 | 1 | 0.00003 | 1 | 1 | 0.000045 | 1 | 1 | 0.000045 | 0.9999 | 1 | 0.00012 |

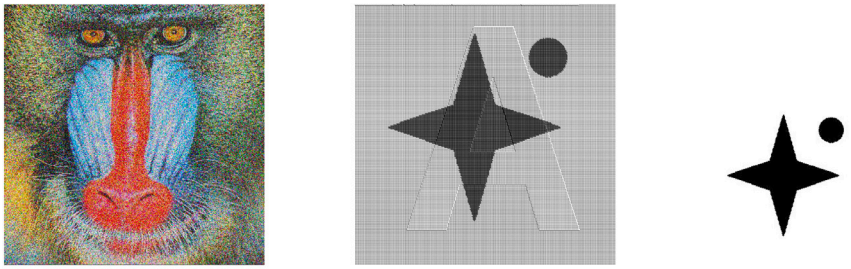



**Fig. 12.** (a) Image with Gaussian Noise ($V$ = 0.09, $PSNR$ = 14.49) (b) Superimposed Result (c) Reduced Watermark ($NC$ = 0.9999).

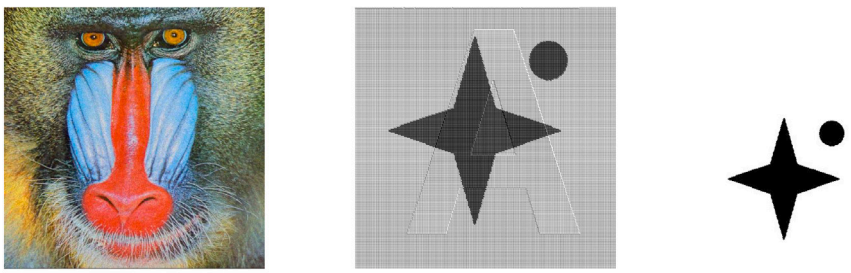



**Fig. 13.** (a) Image with Poisson Noise ($PSNR$ = 27.0710) (b) Superimposed Result (c) Reduced Watermark ($NC$ = 1.00).

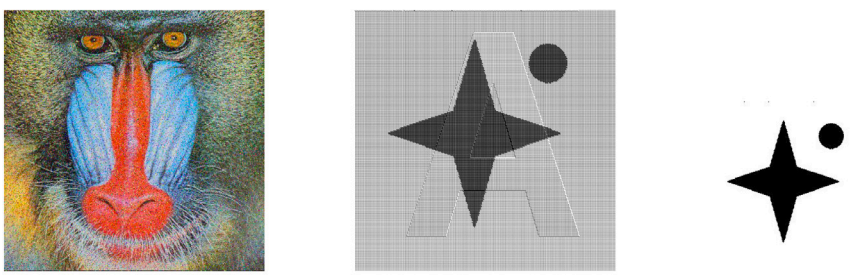



**Fig. 14.** (a) Image with Salt and Pepper Noise ($D$ = 0.09, $PSNR$ = 18.3490) (b) Superimposed Result (c) Reduced Watermark ($NC$ = 1.00).

#### *4.2.2. Structural similarity index measure*

This parameter is used to measure the similarity between two images by calculating similarity for various windows of the image. The similarity measure between two blocks of same size is given by:

$$SSIM(B_1, B_2) = \frac{(2\mu_{B_1}\mu_{B_2} + c_1)(2\sigma_{B_1B_2} + c_2)}{(\mu_{B_1}^2 + \mu_{B_2}^2 + c_1)(\sigma_{B_1}^2 + \sigma_{B_2}^2 + c_2)} \tag{12}$$

where $B_1$ and $B_2$ are the two different blocks, $\mu_{B_1}$ and $\mu_{B_2}$ are the average values of $B_1$ and $B_2$, $\sigma_{B_1}^2$ and $\sigma_{B_2}^2$ are the variance of $B_1$ and $B_2$, $\sigma_{B_1B_2}$ is the covariance of $B_1$ and $B_2$, $c_1$ and $c_2$ are two variables used to stabilize the division where $c_1 = (k_1 L)^2$, $c_2 = (k_1 L)^2$, $L$ is the maximum value the pixels can represent, $k_1 = 0.01$ and $k_2 = 0.03$ taken as default.

#### *4.2.3. Bit error rate*

Robustness of the scheme is measured as BER of the retrieved watermark. It is defined as:

$$BER = \frac{X}{N_s \times N_s} \tag{13}$$

where $X$ represents number of received watermark bits that have been modified due to noise and $N_s \times N_s$ is the size of the watermark image. The range of $BER$ is [0, 1]. The low value of $BER$ indicates high detection performance for a given watermarking scheme.

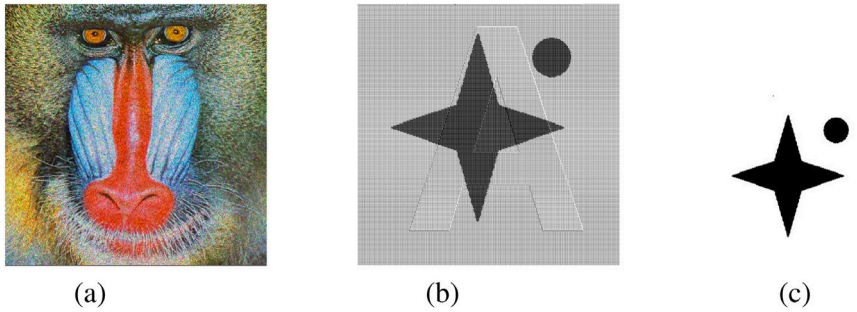



**Fig. 15.** (a) Image with Speckle Noise, *PSNR* = 18.5967) (b) Superimposed Result (c) Reduced Watermark (*NC* = 1.00).

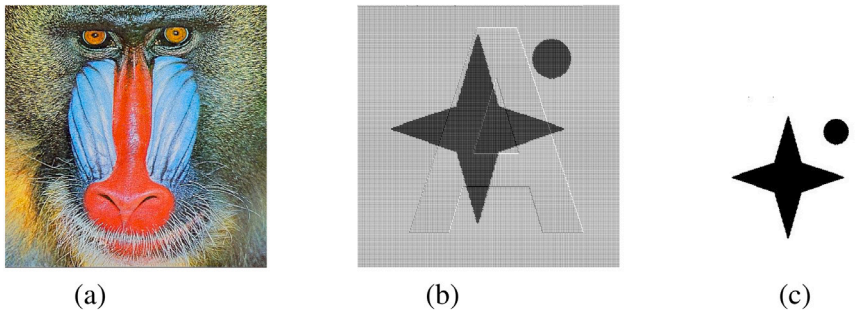



**Fig. 16.** (a) Sharpened Image (*R* = 1, *A* = 2, *PSNR* = 18.2728) (b) Superimposed Result (c) Reduced Watermark (*NC* = 1.00).

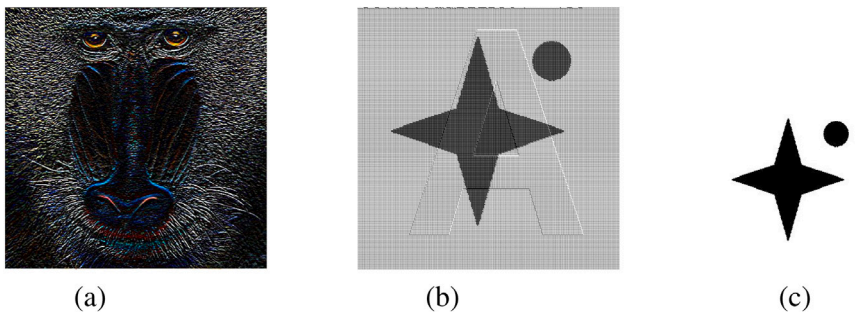



**Fig. 17.** (a) Image with Sobel Attack (*PSNR*=6.3443) (b) Superimposed Result (c) Reduced Watermark (*NC* = 0.9988).

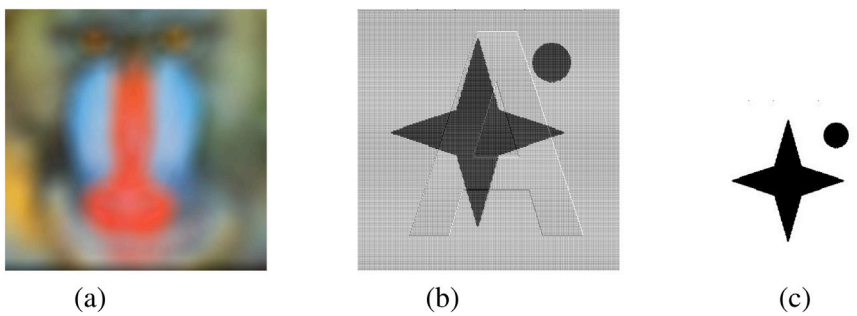



**Fig. 18.** (a) Blurred Image (*R* = 25, *PSNR* = 17.5692) (b) Superimposed Result (c) Reduced Watermark (*NC* = 1.00).

**Table 2**
Comparison of Proposed Scheme with existing schemes for different images and at various attacks.

| Images | | Rawat et al. [8] | Hou et al. [19] | Hou et al. [10] | Devi et al. [11] | Abraham et al. [20] | Roy et al. [21] | Murali et al. [22] | Ernawan et al. [23] | Hurrah et al. [24] | Thanki et al. [25] | Proposed Scheme |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| JPEG Compression | | | | | | | | | | | | |
| Lena | Q=10 | – | 0.59 | – | – | – | – | 0.85 | – | – | – | 1 |
| | Q=20 | – | 0.6 | – | – | 0.7362 | – | 0.85 | – | – | – | 1 |
| | Q=50 | 0.9611 | – | – | 0.9804 | 0.8873 | – | – | – | – | – | 1 |
| | Q=70 | 0.9648 | – | – | 0.9873 | 0.8314 | – | – | – | – | – | 1 |
| | Q=90 | 0.9772 | – | – | 0.9931 | – | – | – | – | – | – | 1 |
| Airplane | Q=5 | – | 0.9943 | 0.9944 | – | – | – | – | – | – | – | 1 |
| | Q=50 | 0.9631 | – | – | 0.9797 | – | – | – | – | – | – | 1 |
| | Q=70 | 0.9697 | – | – | 0.9987 | – | – | – | – | – | – | 1 |
| | Q=90 | 0.9814 | – | – | 0.9934 | – | – | – | – | – | – | 1 |
| Lake | Q=10 | – | 0.6 | – | – | – | – | 0.87 | – | – | – | 1 |
| | Q=20 | – | 0.61 | – | – | – | – | 0.89 | – | – | – | 1 |
| Mandrill | Q=10 | – | 0.6 | – | – | – | – | 0.81 | – | – | – | 1 |
| | Q=20 | – | 0.63 | – | – | – | – | 0.82 | – | – | – | 1 |
| House | Q=10 | – | 0.62 | – | – | – | – | 0.8 | – | – | – | 1 |
| | Q=20 | – | 0.63 | – | – | – | – | 0.81 | – | – | – | 1 |
| Goldhill | Q=50 | 0.9592 | – | – | 0.9946 | – | – | – | – | – | – | 1 |
| | Q=70 | 0.9667 | – | – | 0.9948 | – | – | – | – | – | – | 1 |
| | Q=90 | 0.9765 | – | – | 0.9973 | – | – | – | – | – | – | 1 |
| Peppers | Q=40 | – | 0.97 | – | – | – | 0.6978 | – | 0.661 | – | 0.9597 | 1 |
| | Q=50 | – | – | – | – | – | 0.7502 | – | 0.6382 | – | 0.9639 | 1 |
| | Q=60 | – | – | – | – | – | 0.8067 | – | 0.6087 | – | 0.9741 | 1 |
| | Q=70 | – | – | – | – | – | 0.8916 | – | 0.5876 | – | 0.9837 | 1 |
| | Q=80 | – | – | – | – | – | 0.9404 | – | 0.6069 | – | 0.9898 | 1 |
| | Q=90 | – | – | – | – | – | 0.9639 | – | 0.637 | – | 0.991 | 1 |
| Rotation | | | | | | | | | | | | |
| Lena | A=1 | 0.8549 | – | – | 0.9382 | – | – | – | – | – | – | 0.9999 |
| | A=5 | 0.7353 | – | – | 0.9589 | – | – | – | – | – | – | 0.9999 |
| | A=20 | – | 0.52 | – | – | – | – | 0.79 | – | – | – | 0.9991 |
| Airplane | A=1 | 0.7927 | – | – | 0.9331 | – | – | – | – | – | – | 0.9999 |
| | A=5 | 0.7021 | – | – | 0.9587 | – | – | – | – | – | – | 0.9999 |
| Peppers | A=20 | – | – | – | – | – | 0.5593 | – | 0.5864 | – | – | 0.9991 |
| | A=45 | – | – | – | – | – | 0.5521 | – | – | – | – | 0.9889 |
| Lake | A=20 | – | 0.56 | – | – | – | – | 0.78 | – | – | – | 0.9998 |
| Mandrill | A=20 | – | 0.54 | – | – | – | – | 0.79 | – | – | – | 0.9998 |
| House | A=20 | – | 0.5 | – | – | – | – | 0.8 | – | – | – | 0.9998 |
| Goldhill | A=1 | 0.8493 | – | – | 0.9826 | – | – | – | – | – | – | 0.9999 |
| | A=5 | 0.6992 | – | – | 0.9797 | – | – | – | – | – | – | 0.9999 |
| Median Filtering | | | | | | | | | | | | |
| Lena | $ws = 2 \times 2$ | – | 0.61 | – | – | – | – | 0.95 | – | – | – | 1 |
| | $ws = 3 \times 3$ | 0.9514 | – | – | 0.979 | – | – | – | – | – | – | 1 |
| | $ws = 5 \times 5$ | 0.9379 | – | – | 0.9606 | – | – | – | – | – | – | 1 |
| | $ws = 7 \times 7$ | 0.9279 | – | – | 0.9484 | – | – | – | – | – | – | 1 |
| Lake | $ws = 2 \times 2$ | – | 0.61 | – | – | – | – | 0.87 | – | – | – | 1 |
| Mandrill | $ws = 2 \times 2$ | – | 0.54 | – | – | – | – | 0.89 | – | – | – | 1 |
| House | $ws = 2 \times 2$ | – | 0.63 | – | – | – | – | 0.85 | – | – | – | 1 |
| Airplane | $ws = 3 \times 3$ | 0.9638 | – | – | 0.9873 | – | – | – | – | – | – | 1 |
| | $ws = 5 \times 5$ | 0.9421 | – | – | 0.9724 | – | – | – | – | – | – | 1 |
| | $ws = 7 \times 7$ | 0.925 | – | – | 0.9577 | – | – | – | – | – | – | 1 |

(*continued on next page*)

**Table 2** (*continued*).

| Images | | Rawat et al. [8] | Hou et al. [19] | Hou et al. [10] | Devi et al. [11] | Abraham et al. [20] | Roy et al. [21] | Murali et al. [22] | Ernawan et al. [23] | Hurrah et al. [24] | Thanki et al. [25] | Proposed Scheme |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Goldhill | $ws = 3 \times 3$ | 0.9499 | – | – | 0.9912 | – | – | – | – | – | – | 1 |
| | $ws = 5 \times 5$ | 0.9299 | – | – | 0.9851 | – | – | – | – | – | – | 1 |
| | $ws = 7 \times 7$ | 0.9187 | – | – | 0.9807 | – | – | – | – | – | – | 1 |
| Peppers | $ws = 3 \times 3$ | – | – | – | – | – | 0.8055 | – | – | – | – | 0 |
| | $ws = 5 \times 5$ | – | – | – | – | – | 0.3721 | – | – | – | – | 0 |
| | $ws = 7 \times 7$ | – | – | – | – | – | 0.407 | – | – | – | – | 0.000015 |
| Cropping | | | | | | | | | | | | |
| Lena | %C=20 | 0.8481 | – | – | 0.8261 | – | – | – | – | – | – | 1 |
| | %C=25 | – | – | – | – | 0.75 | – | – | – | 1 | – | 1 |
| | %C=40 | 0.842 | – | – | 0.7897 | – | – | – | – | – | – | 0.9999 |
| | %C=50 | – | 0.79 | 0.92 | 0.7333 | 0.5 | – | – | – | 0.998 | – | 0.9999 |
| | %C=60 | 0.7863 | – | – | – | – | – | – | – | – | – | 0.9998 |
| | %C=75 | – | – | – | – | 0.25 | – | – | – | 0.989 | – | 0.9997 |
| Airplane | %C=20 | 0.7243 | – | – | 0.7333 | – | – | – | – | – | – | 1 |
| | %C=40 | 0.7458 | – | – | 0.7192 | – | – | – | – | – | – | 0.9999 |
| | %C=60 | 0.686 | – | – | 0.7285 | – | – | – | – | – | – | 0.9998 |
| Peppers | %C=11 | – | 0.99 | – | – | – | – | – | – | – | – | 1 |
| | %C=50 | – | 0.8415 | 0.8473 | – | – | 0.9657 | – | – | – | – | 0.9999 |
| Mandrill | %C=50 | – | 0.81 | 0.99 | – | – | – | – | – | – | – | 0.9999 |
| House | %C=50 | – | 0.83 | 0.8 | – | – | – | – | – | – | – | 0.9999 |
| Lake | %C=50 | – | 0.85 | 0.82 | – | – | – | – | – | – | – | 0.9999 |
| Goldhill | %C=20 | 0.7814 | – | – | 0.8969 | – | – | – | – | – | – | 1 |
| | %C=40 | 0.8049 | – | – | 0.8054 | – | – | – | – | – | – | 0.9999 |
| | %C=60 | 0.7177 | – | – | 0.728 | – | – | – | – | – | – | 0.9998 |
| Gaussian Noise | | | | | | | | | | | | |
| Lena | V=0.01 | 0.8374 | 0.53 | – | 0.9318 | – | – | 0.89 | – | – | – | 1 |
| | V=0.03 | 0.7761 | – | – | 0.8981 | – | – | – | – | – | – | 1 |
| | V=0.05 | 0.7556 | – | – | 0.874 | – | – | – | – | – | – | 1 |
| Airplane | V=0.01 | 0.832 | – | – | 0.9267 | – | – | – | – | – | – | 1 |
| | V=0.03 | 0.7663 | – | – | 0.8859 | – | – | – | – | – | – | 1 |
| | V=0.05 | 0.7602 | 0.9085 | 0.9548 | 0.8691 | – | – | – | – | – | – | 1 |
| Mandrill | | – | 0.58 | – | – | – | – | 0.83 | – | – | – | 1 |
| House | | – | 0.55 | – | – | – | – | 0.82 | – | – | – | 1 |
| Lake | | – | 0.54 | – | – | – | – | 0.9 | – | – | – | 1 |
| Goldhill | V=0.01 | 0.8564 | – | – | 0.9755 | – | – | – | – | – | – | 1 |
| | V=0.03 | 0.7902 | – | – | 0.9599 | – | – | – | – | – | – | 1 |
| | V=0.05 | 0.7751 | – | – | 0.9475 | – | – | – | – | – | – | 0.9998 |
| Peppers | V=0.01 | – | – | – | – | – | 0.6412 | – | – | – | – | 1 |
| | V=0.02 | – | – | – | – | – | 0.6075 | – | – | – | – | 1 |
| | V=0.1 | – | – | – | – | – | 0.5569 | – | – | – | – | 1 |
| Sharpening | | | | | | | | | | | | |
| Lena | alpha=0.1 | 0.8688 | 0.75 | – | 0.9208 | 0.9455 | – | 0.87 | – | 0.975 | – | 1 |
| Airplane | alpha=0.1 | 0.8906 | 0.9249 | 0.9255 | 0.9357 | – | – | – | – | – | – | 1 |
| Peppers | alpha=0.1 | – | 0.96 | – | – | – | 0.9645 | – | – | – | – | 1 |
| Lake | alpha=0.1 | – | 0.76 | – | – | – | – | 0.92 | – | – | – | 1 |
| Mandrill | alpha=0.1 | – | 0.79 | – | – | – | – | 0.83 | – | – | – | 1 |
| House | alpha=0.1 | – | 0.79 | – | – | – | – | 0.83 | – | – | – | 1 |

(*continued on next page*)

**Table 2** (*continued*).

| Images | | Rawat et al. [8] | Hou et al. [19] | Hou et al. [10] | Devi et al. [11] | Abraham et al. [20] | Roy et al. [21] | Murali et al. [22] | Ernawan et al. [23] | Hurrah et al. [24] | Thanki et al. [25] | Proposed Scheme |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Goldhill | alpha=0.1 | 0.8737 | – | – | 0.9658 | – | – | – | – | – | – | 1 |
| Lightening | | | | | | | | | | | | |
| Airplane | 20% | – | 0.9943 | 0.9952 | – | – | – | – | – | – | – | 1 |
| Peppers | 20% | – | 1 | – | – | – | – | – | – | – | – | 1 |
| Darkening | | | | | | | | | | | | |
| Airplane | 20% | – | 0.9939 | 0.9939 | – | – | – | – | – | – | – | 1 |
| Peppers | 20% | – | 1 | – | – | – | – | – | – | – | – | 1 |
| Blurring | | | | | | | | | | | | |
| Peppers | | – | 0.99 | – | – | – | 0.5316 | – | – | – | – | 1 |
| Salt and Pepper Noise | | | | | | | | | | | | |
| Lena | | – | 0.58 | – | – | 0.97 | – | 0.89 | – | 0.96 | – | 1 |
| Lake | | – | 0.59 | – | – | – | – | 0.91 | – | – | – | 1 |
| Mandrill | | – | 0.59 | – | – | – | – | 0.81 | – | – | – | 1 |
| House | | – | 0.58 | – | – | – | – | 0.82 | – | – | – | 1 |
| Peppers | | | – | – | – | – | 0.5852 | – | – | – | – | 1 |
| Speckle Noise | | | | | | | | | | | | |
| Lena | | – | – | – | – | 0.91 | – | – | – | 0.97 | – | 1 |
| Poisson Noise | | | | | | | | | | | | |
| Lena | | – | – | – | – | 0.9713 | – | – | – | 0.96 | – | 1 |

The '–' refers that the readings for those cells is not given in the referred works.

#### 4.2.4. Peak signal to noise ratio

Peak Signal to Noise Ratio (*PSNR*) is a parameter used to measure the quality of the image based on the pixel difference between the two images. Higher the value of *PSNR*, better the value of the reconstructed image. It can be defined as:

$$PSNR = 10\log_{10}(\frac{s^2}{MSE})$$

where $s = 2^b - 1$ for every color image, where $b$ is bit depth of the image. *MSE* represents Mean Square Error.

In the proposed scheme, *PSNR* has been used to measure the quality of the attacked image with respect to the original image.

Robustness tests of the proposed scheme on different images against various attacks are shown in Table 1 with respect to *NC*, *BER* and *SSIM*. The image manipulation attacks performed on test images are as follows:

1. ***JPEG* Compression:** The test images have been compressed with quality ($Q$) factor ranging from 40 to 90. *NC* and *BER* of retrieved watermark is above 0.9998 and below 0.00016 respectively, for all test images. *SSIM* of extracted watermark is 1 for all test images. The results for the Mandrill image are shown in Fig. 8, where Fig. 8(a) shows the compressed image with $Q = 90$, $PSNR = 33.45$ dB, Fig. 8(b) shows the superimposed result of *MS* and *OS* while Fig. 8(c) shows the reduced watermark with $NC = 1.00$. This proves the proposed scheme is robust against this attack.
2. **Rotation Attack:** The test images are rotated with angles ($A$) of 10°, 20°, 30°, 40°, 50° and 60°. *NC* and *BER* of retrieved watermark is above 0.9991 and below 0.00091 respectively, for all test images. *SSIM* of extracted watermark is above 0.9999 for all test images. The results for the Mandrill image are shown in Fig. 9, where Fig. 9(a) shows the rotated image by an angle of 60°, $PSNR = 9.4488$ dB, Fig. 9(b) shows the superimposed result of *MS* and *OS* while Fig. 9(c) shows the reduced watermark with $NC = 0.9993$. This proves the proposed scheme is robust against this attack, even when the attacked image has remarkably low *PSNR*.
3. **Median Filtering Attack:** The test images are applied filtering attack for window sizes ($ws$) of $2\times2$, $3\times3$, $4\times4$, $5\times5$, $6\times6$ and $7\times7$. *NC* and *BER* of retrieved watermark is above 0.9999 and below 0.0003 respectively, for all test images. *SSIM* of extracted watermark is 1 for all test images. The results for the Mandrill image are shown in Fig. 10, where Fig. 10(a) shows the image with Median filtering attack for a window size of $3\times3$, $PSNR = 20.4138$ dB, Fig. 10(b) shows the superimposed result of *MS* and *OS* while Fig. 10(c) shows the reduced watermark with $NC = 1.00$. This proves the proposed scheme is robust against this attack.
4. **Cropping Attack:** The test images are cropped for window size $16\times16$, $32\times32$, $64\times64$, $128\times128$, and $256\times256$. *NC* and *BER* of retrieved watermark is above 0.9991 and below 0.00093 respectively, for all test images. *SSIM* of extracted watermark is above 0.9999 for all test images. The results for the Mandrill image are shown in Fig. 11, where Fig. 11(a) shows the cropped image for a window size of $128\times128$, $PSNR = 18.0368$ dB, Fig. 11(b) shows the superimposed result of *MS* and *OS*; Fig. 11(c) shows the reduced watermark with $NC = 0.9997$. This proves the proposed scheme is robust against this attack.
5. **Gaussian Noise Attack:** The gaussian noise is added to the test images with mean 0 and variance ($V$) ranging from 0.01 to 0.09. *NC* and *BER* of retrieved watermark is above 0.9997 and below 0.00076 respectively, for all test images. *SSIM* of extracted watermark is 1 for all test images. The results for the Mandrill image are shown in Fig. 12, where Fig. 12(a) shows the image attacked by Gaussian Noise with $V = 0.09$, $PSNR = 14.49$ dB, Fig. 12(b) shows the superimposed result of *MS* and *OS*; Fig. 12(c) shows the reduced watermark with $NC = 0.9999$. This proves the proposed scheme is robust against this attack.
6. **Poisson noise:** The noise is added from the image data itself instead of adding artificial noise to the image. *NC*, *BER* and *SSIM* of retrieved watermark is 1, 0 and 1 respectively, for all test images. The results for the Mandrill image are shown in Fig. 13, where Fig. 13(a) shows the image with Poisson Noise having $PSNR = 27.0710$ dB, Fig. 13(b) shows the superimposed result of *MS* and *OS*; Fig. 13(c) shows the reduced watermark with $NC = 1.00$. This proves the proposed scheme is robust against this attack.
7. **Salt and Pepper Noise:** This noise is added to the test images with noise density ($D$) from 0.01 to 0.09. *NC* and *BER* of retrieved watermark is above 0.9998 and below 0.0001 respectively, for all test images. *SSIM* of extracted watermark is 1 for all test images. The results for the Mandrill image are shown in Fig. 14, where Fig. 14(a) shows the image with Salt and Pepper Noise having $PSNR = 18.3490$ dB, Fig. 14(b) shows the superimposed result of *MS* and *OS*; Fig. 14(c) shows the reduced watermark with $NC = 1.00$. This proves the proposed scheme is robust against this attack.
8. **Speckle Noise:** This noise is added to all the test images with variance ($V$) from 0.01 to 0.10. *NC* and *BER* of retrieved watermark is above 0.9999 and below 0.00003 respectively, for all test images. *SSIM* of extracted watermark is 1 for all test images. The results for the Mandrill image are shown in Fig. 15, where Fig. 15(a) shows the image with Speckle Noise having $PSNR = 18.5967$ dB, Fig. 15(b) shows the superimposed result of *MS* with *OS*; Fig. 15(c) shows the reduced watermark with $NC = 1.00$. This proves the proposed scheme is robust against this attack.
9. **Sharpening Attack:** The test images are sharpened with different combinations of radius ($R$) and amount ($A$). *NC* and *BER* of retrieved watermark is above 0.9999 and below 0.000091 respectively, for all test images. *SSIM* of extracted watermark is 1 for all test images. The results for the Mandrill image are shown in Fig. 16, where Fig. 16(a) shows a sharpened image with $PSNR = 18.2728$ dB, Fig. 16(b) shows the superimposed result of *MS* and *OS*; Fig. 16(c) shows the reduced watermark with $NC = 1.00$. This proves the proposed scheme is robust against this attack.
10. **Sobel Attack:** This attack returns an image with the edges detected. *NC* and *BER* of retrieved watermark is above 0.9983 and below 0.0023 respectively, for all test images. *SSIM* of extracted watermark is above 0.9999 for all test images. The results for the Mandrill image are shown in Fig. 17, where Fig. 17(a) shows the image with Sobel Attack having $PSNR = 6.3443$ dB,

Fig. 17(b) shows the superimposed result of *MS* and *OS*; Fig. 17(c) shows the reduced watermark with *NC* = 0.9988. This proves the proposed scheme is robust against this attack.

11. **Blurring attack:** The images are blurred for different values of radius (*R*). *NC* and *BER* of retrieved watermark is above 0.9999 and below 0.00012 respectively, for all test images. *SSIM* of extracted watermark is 1 for all test images. The results for the Mandrill image are shown in Fig. 18, where Fig. 18(a) shows the blurred image having *PSNR* = 17.5692 dB, Fig. 18(b) shows the superimposed result of *MS* and *OS*; Fig. 18(c) shows the reduced watermark with *NC* = 1.00. This proves the proposed scheme is robust against this attack.

These results are presented in Table 1 from where it can be observed that performance of the proposed scheme is outstanding against all attacks on different images for various ranges. The values for *NC* in all the cases are above 0.99. *BER* values are close to 0 for all test images. This proves high robustness of the scheme.

Comparison of the proposed scheme with existing schemes on different images against different attacks is shown in Table 2.

It can be observed from Table 2 that for the existing schemes, *NC* for different images against *JPEG* compression attack ranges from 0.59 to 0.9944, while it is 1 for the proposed scheme. For rotation, it ranges from 0.52 to 0.9826 in the existing scheme, while for the proposed scheme it remains 0.99. For median filtering, it ranges from 0.3721 to 0.9873 for existing schemes and is 1 for the proposed scheme. For Cropping and Gaussian Noise, it ranges from 0.25 to 0.998 and 0.53 to 0.9755, respectively in the existing schemes, while for the proposed scheme it ranges between 0.99 and 1. For sharpening, it ranges from 0.75 to 0.975 for existing schemes but is 1 for the proposed scheme. For lightening, darkening and blurring, it ranges between 0.5316 and 1 for the existing schemes, while for the proposed scheme it is 1. For salt and pepper noise, it ranges from 0.58 to 0.97 and is 1 for the proposed scheme. The use of *LF* and *MF* sub-bands of *DCuT* coefficients help in enhancing the robustness of the scheme.

### *4.3. Security analysis*

In this scheme, the codebook has been designed to create meaningful shares that are stored with the *TA*. These meaningful shares help in enhancing security of the scheme, as these are visually similar to the cover images and do not create any suspicion that some secret information is being shared or stored.

Secondly, before creating shares, the watermark and coefficient matrices are pre-processed through Baker's Map which scrambles the image, thereby securing the scheme.

## 5. Conclusion

Multimedia data and medical images are often applied to devices with low computation power. Due to computational constraints, watermarking schemes with less complexity are required. Thus, in this paper, a new secure and robust copyright protection scheme based on DCuT, *k*-means Clustering and *EVC* is proposed. The selection of non fine scale layer DCuT coefficients enhance the robustness of the scheme. Baker's Map is used for scrambling host image and watermark to make the scheme secure. A codebook is proposed to create meaningful ownership shares, thereby ensuring security of the scheme. The complexity of scheme is very low as watermark can be retrieved blindly just by *XOR*-superimposition of the shares. Experiments have been performed on different images by doing different attacks to check robustness of the scheme. *NC* value of the extracted watermark is maintained at 0.99 or more, which shows that the scheme has outstanding resistance to attacks. The advantages of the proposed scheme are high robustness, imperceptibility, security and blind detection. The watermark's size is not restricted to the size of the protected image. Comparison with the state-of-art copyright protection schemes reveals that the proposed scheme gives better performance. This scheme can be extended to multiple color images with multiple owners.

## Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.compeleceng.2020.106931.

## References

[1] Naor Moni, Shamir Adi. Visual cryptography. In: Workshop on the theory and application of of cryptographic techniques. Springer; 1994, p. 1–12.
[2] Pahuja Shivani, Kasana Singara Singh. Halftone visual cryptography for color images. In: 2017 international conference on computer, communications and electronics (Comptelix). IEEE; 2017, p. 281–5.
[3] Hou Young-Chang, Chen Pei-Min. An asymmetric watermarking scheme based on visual cryptography. In: Signal processing proceedings, 2000. WCCC-ICSP 2000. 5th international conference on, vol. 2. IEEE; 2000, p. 992–5.
[4] Hwang Ren-Junn. A digital image copyright protection scheme based on visual cryptography. Tamkang J Sci Eng 2000;3(2):97–106.
[5] Hassan Mahmoud A, Khalili Mohammed A. Self watermarking based on visual cryptography. In: Proceedings of world academy of science, engineering and technology, vol. 8; 2005. p. 159–62.
[6] Hsu Ching-Sheng, Hou Young-Chang. Copyright protection scheme for digital images using visual cryptography and sampling methods. Opt Eng 2005;44(7):077003.
[7] Wang Ming-Shi, Chen Wei-Che. A hybrid DWT-SVD copyright protection scheme based on k-means clustering and visual cryptography. Comput Stand Interfaces 2009;31(4):757–62.
[8] Rawat Sanjay, Raman Balasubramanian. A publicly verifiable lossless watermarking scheme for copyright protection and ownership assertion. AEU-Int J Electron Commun 2012;66(11):955–62.

[9] Chen Tzung-Her, Chang Chin-Chen, Wu Chang-Sian, Lou Der-Chyuan. On the security of a copyright protection scheme based on visual cryptography. Comput Stand Interfaces 2009;31(1):1–5.
[10] Hou Young Chang, Tseng A-Yu, Quan Zen-Yu, Liu Hsin-Ju. An IPR protection scheme based on wavelet transformation and visual cryptography. Turkish J Electr Eng Comput Sci 2016;24(5):4063–82.
[11] Devi B Pushpa, Singh Kh Manglem, Roy Sudipta. A copyright protection scheme for digital images based on shuffled singular value decomposition and visual cryptography. SpringerPlus 2016;5(1):1091.
[12] Shao Zhuhong, Shang Yuanyuan, Zeng Rui, Shu Huazhong, Coatrieux Gouenou, Wu Jiasong. Robust watermarking scheme for color image based on quaternion-type moment invariants and visual cryptography. Signal Process, Image Commun 2016;48:12–21.
[13] Xue Mingfu, Yuan Chengxiang, Liu Zhe, Wang Jian. SSL: A novel image hashing technique using SIFT keypoints with saliency detection and LBP feature extraction against combinatorial manipulations. Secur Commun Netw 2019;2019.
[14] Fatahbeygi Ali, Tab Fardin Akhlaghian. A highly robust and secure image watermarking based on classification and visual cryptography. J Inf Secur Appl 2019;45:71–8.
[15] Candès Emmanuel J, Donoho David L. New tight frames of curvelets and optimal representations of objects with piecewise C2 singularities. Comm Pure Appl Math: J Issued Courant Inst Math Sci 2004;57(2):219–66.
[16] Fox Ronald F. Construction of the Jordan basis for the baker map. Chaos 1997;7(2):254–69.
[17] Driebe Dean J. Fully chaotic maps and broken time symmetry, vol. 4. Springer Science & Business Media; 1999.
[18] Agarwal Shafali. A review of image scrambling technique using chaotic maps. Int J Eng Technol Innov 2018;8(2):77–98.
[19] Hou Young-Chang, Huang Pei-Hsiu. An ownership protection scheme based on visual cryptography and the law of large numbers. Int J Innov Comput 2012;(6):4147–56.
[20] Abraham Jobin, Paul Varghese. An imperceptible spatial domain color image watermarking scheme. J King Saud Univ-Comput Inf Sci 2016.
[21] Roy Soumitra, Pal Arup Kumar. A robust blind hybrid image watermarking scheme in RDWT-DCT domain using arnold scrambling. Multimedia Tools Appl 2017;76(3):3577–616.
[22] Murali P, Sankaradass Veeramalai. An efficient ROI based copyright protection scheme for digital images with SVD and orthogonal polynomials transformation. Optik 2018;170:242–64.
[23] Ernawan Ferda, Kabir Muhammad Nomani. A block-based RDWT-SVD image watermarking method using human visual system characteristics. Vis Comput 2018;1–19.
[24] Hurrah Nasir N, Parah Shabir A, Loan Nazir A, Sheikh Javaid A, Elhoseny Mohammad, Muhammad Khan. Dual watermarking framework for privacy protection and content authentication of multimedia. Future Gener Comput Syst 2019;94:654–73.
[25] Thanki Rohit, Kothari Ashish, Trivedi Deven. Hybrid and blind watermarking scheme in DCuT–RDWT domain. J Inf Secur Appl 2019;46:231–49.

**Sonal Kukreja** graduated in 2012 from Rajasthan Technical University, Kota, India and received her M.Tech. degree in 2015 in the field of Computer Science and Applications from Thapar Institute of Engineering and Technology, Patiala, India. In 2016, she enrolled herself in Ph.D., at Thapar Institute of Engineering and Technology, Patiala, India. Her research interest is focused on Information Security particularly on data hiding in images, visual cryptography and watermarking.

**Geeta Kasana** is working as Assistant Professor in Computer Science and Engineering Department, Thapar University, Patiala, India. She has ten years of teaching and research experience. She received her Ph.D. degree in information security from Thapar Institute of Engineering and Technology. Her research interests include image processing and information security. She has published many research papers in reputed International Journals and Conferences. She is currently guiding Ph.D. students on Information Security.

**Singara Singh Kasana** is working as Associate Professor in Computer Science and Engineering Department, Thapar Institute of Engineering and Technology, Patiala, Punjab, India. He has eighteen years of teaching and research experience. He received Ph.D. degree in image compression from Thapar Institute of Engineering and Technology. His research interests include image processing, wireless networks, and information security. He has published many research papers in reputed International Journals and Conferences. He is currently guiding Ph.D. students on Information Security, Image and Video Watermarking, Cryptography and Remote Sensing.