IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

# A Robust Multi-Channel Digital Image Watermarking Technique with SVD, DWT, DCT

**Foram R. Suratwala[1*], S. G. Kejgir[2]**

[1,2]Electronics & Tele. Engg. Dept., Shri Guru Gobind Singhji Institute of Engineering and Technology, Vishnupuri, Nanded Maharashtra, India

DOI: https://doi.org/10.26438/ijcse/v7i5.739751 | Available online at: www.ijcseonline.org

Accepted: 25/May/2019, Published: 31/May/2019

***Abstract*—**in this paper, watermark is to hide information in cover image. This paper represents a reliable digital image watermarking. It provides high imperceptibility and robustness for copyright protection. In this proposed method, combination of Discrete Wavelet Transform, Singular Value Decomposition, and Discrete Cosine Transform are used for robust watermarking. A multi-channel image watermarking technique is used in this work. It means multiple watermarks are embedded. In this method, initially first level of three watermark images are embedded in the second RGB watermark image and then this second level RGB image watermark is embedded in the RGB cover image to obtain the watermarked image. This process increases the robustness of watermark. An embedded watermark is retrieved by using reverse process of embedding. The second level of watermark is retrieved and then first levels of watermarks are retrieved. Experimental results show that, as compared to existing method, it shows that proposed multi-channel watermarking method is highly robust against attacks. The experiment result having high Peak Signal to Noise Ratio after different types of attacks. Bit Error Rate and normalized Cross-Correlation shows fidelity of retrieved watermark is very good.

***Keywords*—**Embedding process, extraction process, Peak signal to noise ratio, normalized cross correlation and Bit Error Rate.

## I. INTRODUCTION

The term watermark was likely developed from the German term “wassermarke”. In concern to the growth of high-speed computer networks and that of the internet, digital media offer several advantages over analog media like high quality, easy editing, high fidelity, copying. In this paper, watermark is digital information like digital logo, legal information, certification of ownership of company. Key is used to improve the security and more robustness. If the watermarking is robust then there is no piracy or image processing should not affect the embedded watermark. In this paper, a robust watermarking technique, in which modification to the watermarked content will not affect the watermark. The purpose of a digital watermark is to provide copyright for intellectual property rights. A watermarking is usually divided into four distinct parts: the first part is the watermark embedding process, the second part is extraction of the watermark, the third part is a different attacking process, and the fourth part is after attacks the original watermark is retrieved to check how robust the watermarks. Watermarking techniques are categorized into spatial domain and frequency domain. In the spatial domain, robustness is a major issue as it is easy to identify embedded watermark. In this, watermarks are embedded directly in the image pixel. But in the frequency domain, image is transformed into the frequency domain by using different transforms such as Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), Singular Value Decomposition (SVD) and Discrete Fourier Transform (DFT) that provides robust watermarking.

The outlines of this paper are discussed follow. In section 1 consists of a watermarking introduction. In section 2 consists of literature survey is using related work. In section 3 consists of methodology to given SVD, DWT, DCT are discussed. In section 4 describe the proposed method for embedding process and extraction process. In section 5 describe the experiment result of the Peak Signal to Noise Ratio (PSNR) test using a different type of attack are disused. In section 6 consists of a conclusion.

## II. LITERATURE SURVEY

In Literature survey existing methods are discussed. Nazir A. Loani and *et.al* [1], worked on the medical image, watermarked image is generated by using the DCT algorithm, Arnold, and chaos encryption. Lamri Laouamer and Omar Tayan [2], proposed a method for natural textual based images, in this linear interpolation and tamper detection, are used for a robust non-blind watermark algorithm.In this Inverse Discrete Wavelet Transform (IDWT) is used, resulting in low BER and high Correlation Coefficient (CC). Ferad Ernawan and Muhammad Kabir [3], proposed a method using DCT psych visual thresholding worked on the color image of size m×n, which is divided into 8×8 non-imbrications block after computing entropy non-overlap block and then allot secret zigzag sequence of the cover image. Use this sequence to hide the watermark image. This is the scheme of embedding watermarked technique and scrambled images. Patric Schuch [4], worked on fingerprint images using National institute of standards and technology of USA quality (NFIQ) and score for dataset FVC2000 DB1. They extracted the feature for image enhancement and the curve is used for biometric compression as well as a quality measure. Yahiya Al-Nabhani and *et.al* [5] proposed the method based on discrete wavelet transform and Probabilistic Neural Network (PNN) technique, they have taken a color image then performed wavelet decomposition up to three levels. PNN is supervised learning of neural network implemented bays approach for pattern classification. Amit Phadikarand *et.al* [6], proposed a method based on quard-tree decomposition, in this method, different levels of resolution are varying it is very difficult to merge fragments.

## III. METHODOLOGY

### 1.1 Singular Value Decomposition

SVD is singular value decomposition. Generally, SVD of matrix A is represented as

$$A=USV^T \quad (1)$$

Where $U=[u_1 \cdots u_n]$, $S=\begin{bmatrix}\sigma_1 & 0 & 0\\ 0 & \ddots & 0\\ 0 & 0 & \sigma_n\end{bmatrix}$, $V=\begin{bmatrix}v_1^T\\ \vdots\\ v_n^T\end{bmatrix}$

SVD is a linear algebra transform that is used for the factorization of a real or complex matrix. This is having numerous applications in various field of image processing. As represented in a matrix from which is giving the entire intensity value of each pixel in the image. The meaning of the above equation 1is given below.

1. U is a column orthogonal matrix of size m × n,
2. S is a diagonal matrix with positive or zero elements of size n × n, and
3. V is an orthogonal matrix of size n × n.

Let $S=dig\{\sigma_1, \sigma_2, \sigma_3, \cdots \sigma_n\}$, where, $\sigma_1 \geq \sigma_2 \cdots \sigma_n \geq 0$, and then $\sigma_1, \sigma_2, \sigma_3, \cdots \sigma_n$ are called as the singular values of A.

### 1.2 Discrete Wavelet Transform

DWT was presently used in signal development applications, then audio and video brevity, removing of noise in audio and planetarium of wireless antenna apportionment. DWT is process by image is divide into four part after LL is again dividing, as shown in in fig. (1).

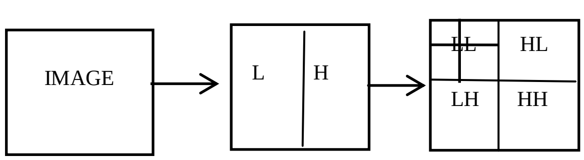



Figure1: Block diagram of DWT

### 1.3 Discrete Cosine Transform

DCT is a simple algorithmic rule of image processing for operations like a low pass filter, brightness, contrast adjustment, and blur, etc. DCT is represented in word frequency space, block diagram of DCT as shown in fig. (2) and represented in mathematical form in equation 2 as follow.

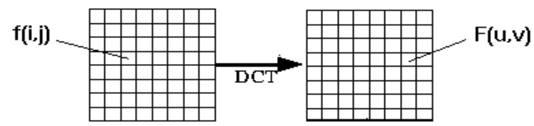



Figure2: Block diagram of DCT

$$F(u,v) = p(u)p(v) \sum_{i=0}^{n-1}\sum_{j=0}^{m-1} f(i,j)\, cos\left(\frac{(2i+1)u\pi}{2n}\right) cos\left(\frac{(2j+1)v\pi}{2m}\right) \quad (2)$$

The input image is size of n and m.

F (i, j) is the intensity of the pixel in row i and column j.

Where, $p(u) = \sqrt{\frac{1}{i}} \quad if \;\; u = 0$

$\sqrt{\frac{2}{i}} \quad if \; u \neq 0$

Similarity, $p(v) = \sqrt{\frac{1}{j}} \quad if \; v = 0$

$\sqrt{\frac{2}{j}} \quad if \; v \neq 0$

## IV. PROPOSED METHOD

Digital image watermarking is one of the most important research areas to provide more security. In this work, a highly robust algorithm is proposed. SVD and scaling factor applied on the First level (Red, Green, and Blue) RGB watermark image, three gray level watermark images are used. Applied on cover RGB image and then second level watermarked RGB images to separate three equal part of Red, Green, Blue section are applied DWT and DCT working with scaling factor. This method is highly robust. It is in the frequency domain.

### 4.1. Watermark Embedding Process:

Step 1:

1. Read first level image and cover images size are 512×512 and three watermark images two watermark images are binary images and third watermark image is RGB image, so three watermark images size are 50× 50 (binary watermark image), 60 × 60 (RGB watermark image) and 32×32 (binary watermark image) as shown in fig. (3).

2. Process of fidelity embedding watermark as follows.

3. SVD applied on first level RGB watermark images are three channels (Red, Green, and Blue) section to get separate different U, S and V matrix.

4. Initially, SVD applied in the first level watermark for two binary watermarks and one RGB image on the blue channel to get different U, S and V matrix.

5. The singular value of three watermark images embedding into singular values of the red, green, blue section of the first level watermark with a key or scaling factor α to obtain new singular value.

6. The next inverse singular value decomposition (ISVD) is performed.

7. Finally, the first level watermarked image is obtained by concatenation of three new red, green, and blue sections as shown in fig. (3).

Step 2:

8. One level DWT method in cover RGB image on blue component to get four coefficients (LL, LH, HL, HH) to get also perform DCT on LH, HL, HH coefficients to obtained new three DCT coefficient values (dc1, dc2, dc3) after combine DWT coefficient of LL so given new component of DWT .
9. One level DWT on each red, green, blue component of second level RGB watermark image to get four coefficients and performances DCT after combine the DWT coefficient, then given a new component of DWT.

10. The DWT and DCT combine of cover RGB image of embedding into DWT and DCT combine of second level RGB watermark image with a key or scaling factor α to obtain new value. I.e. α is 0.002.
11. After this process performs IDWT and IDCT to obtained new blue section.

12. Embedding watermark image with color reconstructed of new blue section with unmodified red and green section of cover RGB image respectively as shown in figure (3).

13. Finally obtained a watermarked image. So, increases more robustness as shown in fig. (3).

**4.2. Watermark Retrieval Process:**

14. Taking the reverse operation process of Step 2 to get second level retrieved watermark image shown in fig. (4).

15. The Second level retrieved watermark image after taking the reverse operation of step 1 points 3, 4, 5, 6 to get other retrieved watermark images as shown in fig. (4).

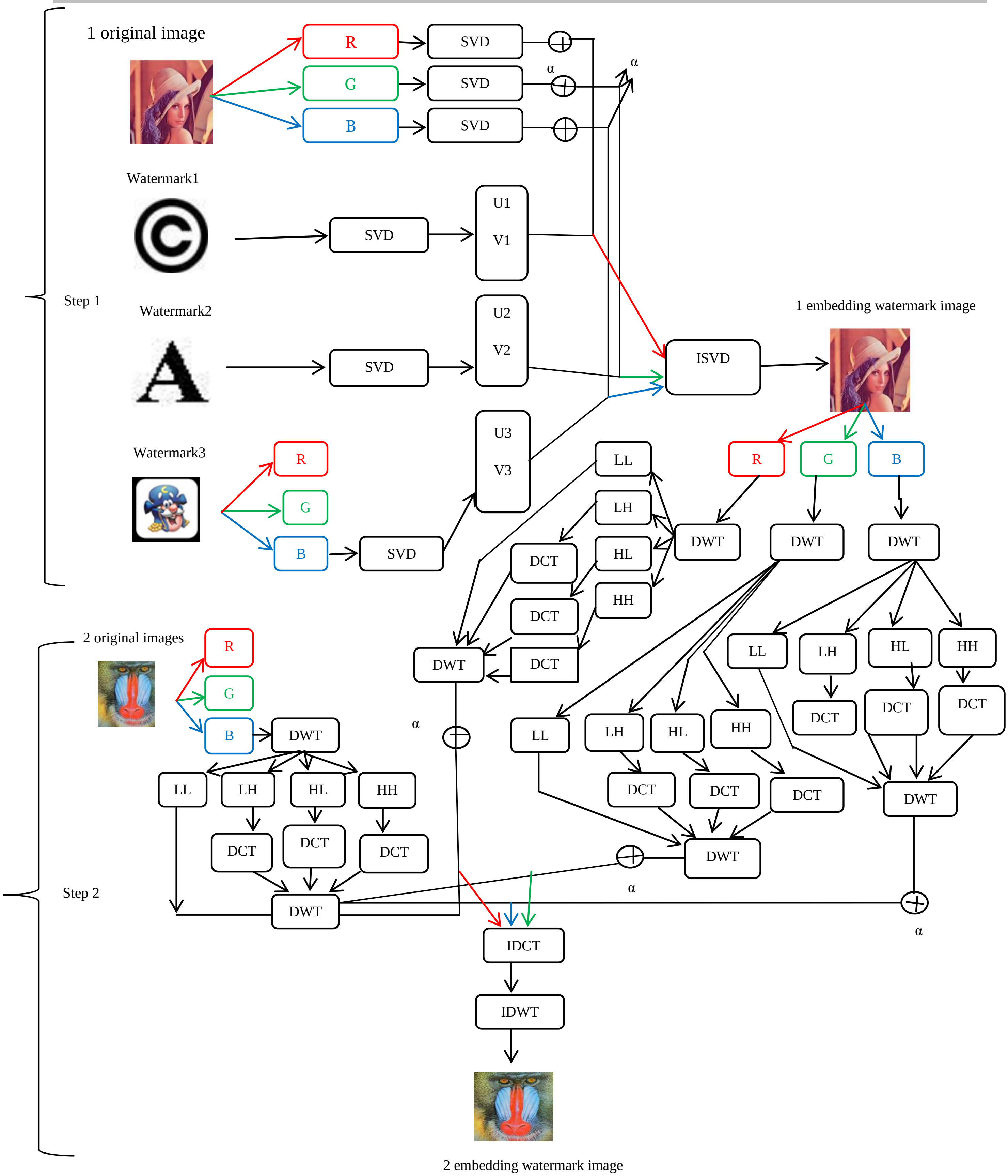



Figure3. Block Diagram of watermark embedding process

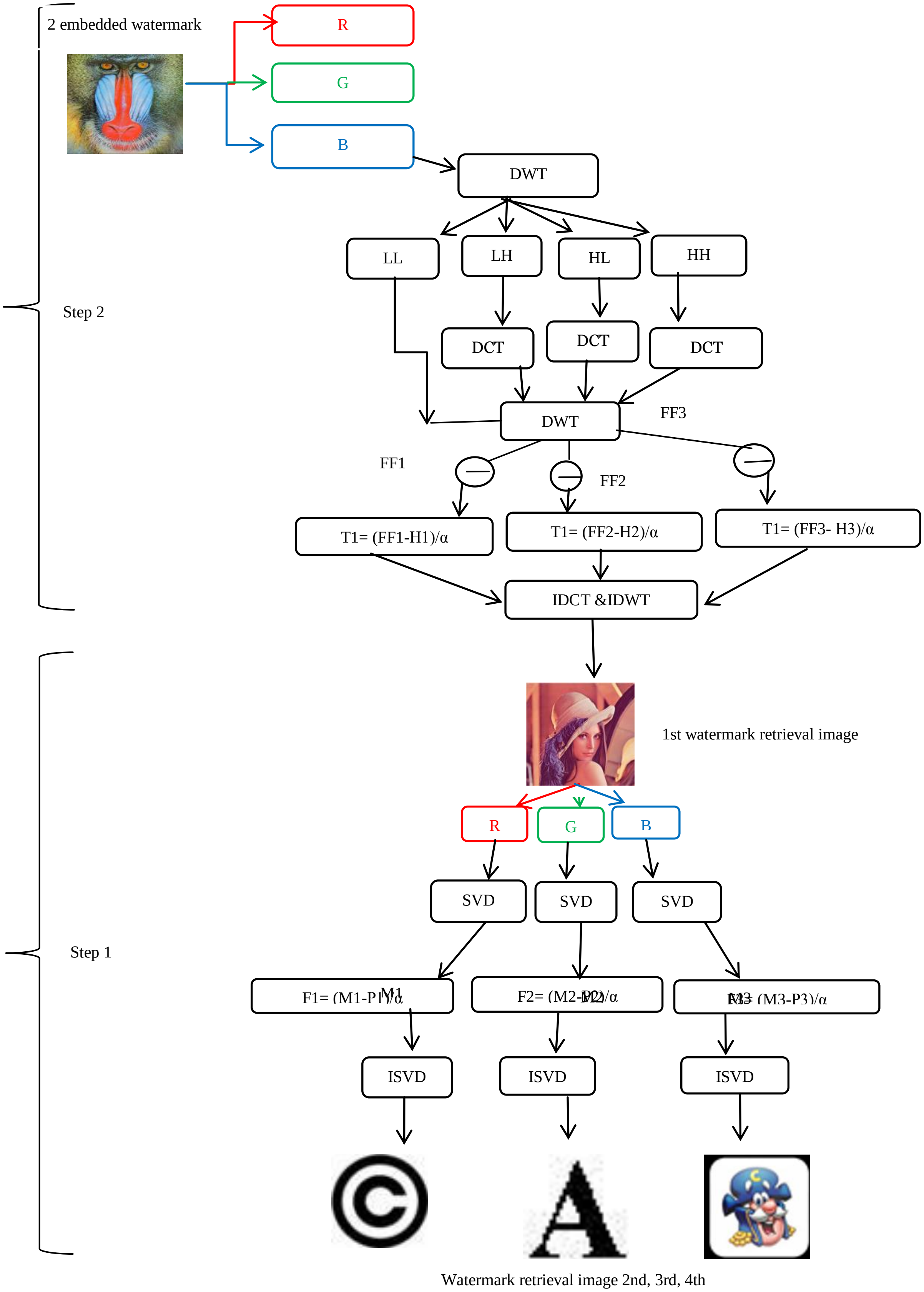



Figure4. Block Diagram of watermark retrieval

## V. EXPERIMENT RESULT

In this experiment, different types of logo images, size with 60×60 (gray image), 50×50 (binary image), 32×32 (binary image) pixels and first level watermark and cover images, size 512×512 pixel. The performance of our technique is evaluated by imperceptibility and robustness. The simulation results were done by benchmark standard "Stir Mark 4.0" software.

(a) (b) (c)

(d) (e) (f)

Figure 5 Original images(a), (b) and (c), Watermark images (d), (e) and (f).

Fig. 5 [a, d, c] shows that original images, fig. 5 [b, e, f] shows that watermark images. The quality of the proposed method is tested by using standard the parameter like Peak Signal Noise Ratio (PSNR) "Stir-Mark 4.0" benchmark software is used [7].

### 5.1. PSNR Test

PSNR in loss image and video compression are between 30 and 50 dB provided for 8 bits, where higher bit is better. For 16-bit data of PSNR are value between 60 and 80 dB, Where, R is 255. Performance result of original images and after watermark embedded images presented in the form of PSNR value, for Lena image it is 63.255, for the baboon image is 62.320, and for air image is 63.555.

$$\text{PSNR} = 10\log_{10}\left(\frac{R^2}{MSE}\right) \quad (3)$$

#### 5.1.1. Affine attack

Affine transformation is an important class of linear 2-D geometric transformation which maps variable is pixel intensity value located at position (X, Y) in an input image into a new variable (X1, Y1) in an output. Images by applying a linear combination of translation, rotation, scaling or shearing operator perform. Fig. 6(a) shows that 2nd watermarked image after affine1 attack, fig. 6(b) shows that 1st watermarked image after affine1 attack, fig. 6(c) is retrieved watermark image 1st after affine1 attack, fig. 6(d) is retrieved watermark image 2nd after affine1 attack, fig. 6(e) is retrieved watermark image 3rd after affine1 attack by using "Stir Mark 4.0". Table-1(a) shows PSNR value after an affine attack using 1st watermarked image, Table-1(b) shows PSNR value after an affine attack using 2nd watermarked images.

(a) (b) (c)

(d) (e)

Figure6. attack of after affine1; (a) 2nd Watermarked image, (b) 1st Watermarked image, (c) retrieved watermark image 1st, (d) retrieved watermark image2nd, (e) retrieved watermark image3rd.

Table 2: (a) for PSNR value of watermarked image 1st & (b) for PSNR value of watermarked image 2nd for affine attack.

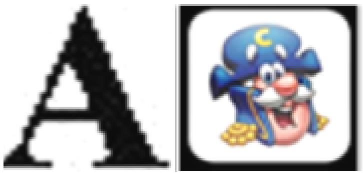
| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| Affine1 | 25.5291 | 22.056 |
| Affine2 | 22.4668 | 19.330 |
| Affine3 | 25.517 | 22.181 |
| Affine4 | 21.742 | 19.500 |
| Affine5 | 24.851 | 21.654 |
| Affine6 | 24.669 | 21.769 |
| Affine7 | 24.921 | 21.769 |
| Affine8 | 25.119 | 21.815 |

#### 5.1.2. Additional noise (PSNR) Attack

It represents original watermark image and watermark image. Fig. 7(a) shows that 2nd Watermarked image after additional noise (PSNR 50) attack, fig. 7(b) as shows 1st watermarked image after additional noise (PSNR 50) attack, fig. 7(c) is retrieved watermark image 1st after additional noise (PSNR 50) attack, fig. 7(d) is retrieved watermark image 2nd after additional noise (PSNR 50) attack, fig. 7(e) is retrieved watermark image 3rd after additional noise (PSNR 50) attack by using "Stir Mark 4.0". Table-2(a) shows PSNR value after additional noise (PSNR) attack using 1st watermarked image, Table-2(b) shows PSNR value after additional noise (PSNR) attack using 2nd watermarked images.

(a) (b) (c)



(d) (e)

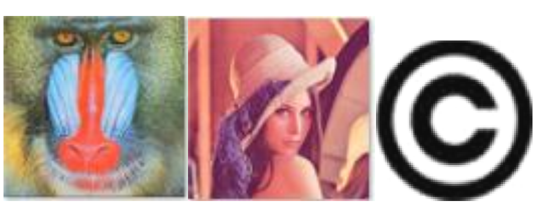
Figure 7 after of additional noise (PSNR 50) attack; (a)$2^{nd}$watermarked image, (b)$1^{st}$watermarked image, (c) retrieved watermark image $1^{st}$, (d) retrieved watermark image $2^{nd}$, (e) retrieved watermark image$3^{rd}$.

Table 2: (a) for PSNR value of watermarked image $1^{st}$ & (b) for PSNR value of watermarked image$2^{nd}$ for add PSNR attack.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| PSNR10 | 30.665 | 25.674 |
| PSNR20 | 30.303 | 25.286 |
| PSNR30 | 29.513 | 25.003 |
| PSNR40 | 28.889 | 24.762 |
| PSNR50 | 27.9007 | 24.341 |
| PSNR60 | 27.41 | 24.031 |
| PSNR70 | 26.290 | 23.537 |
| PSNR80 | 25.690 | 23.197 |
| PSNR90 | 24.848 | 22.678 |
| PSNR100 | 24.321 | 22.331 |

### 5.1.3.JPEG compression Attack

JPEG is currently one of the most widely used compression algorithm for images and the system shows that ‘’quality factor’’. It uses JPEG extensively as well as resizing. Fig. 8(a) shows that $2^{nd}$ watermarked image after JPEG compression 60 attack, fig. 8(b) shows that $1^{st}$ watermarked image after JPEG compression 60 attack, fig. 8(c) is retrieved watermark image $1^{st}$ after JPEG compression 60 attacks, fig. 8(d) is retrieved watermark image $2^{nd}$ after JPEG compression 60 attacks, fig. 8(e) is retrieved watermark image $3^{rd}$ after JPEG compression 60 attacks by using “Stir Mark 4.0”. Table-3(a) shows PSNR value after JPEG compression 60 attacks using the $1^{st}$ watermarked image, Table-3(b) shows PSNR value after JPEG compression 60 using $2^{nd}$ watermarked images.



(a) (b) (c)

(d) (e)

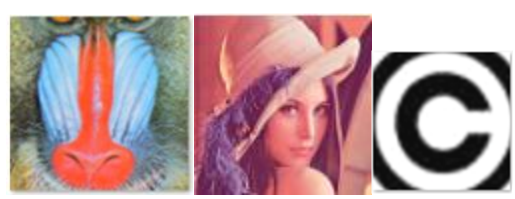
Figure 8 attack of after JPEG compression 60; (a) $2^{nd}$ watermarked image, (b)$1^{st}$ watermarked image, (c) retrieved watermark image $1^{st}$, (d) retrieved watermark image $2^{nd}$, (e) retrieved watermark image$3^{rd}$.

Table 3: (a) for PSNR value of watermarked image $1^{st}$ & (b) for PSNR value of watermarked image $2^{nd}$ for JPEG compression attack.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| JPEG15 | 27.493 | 24.236 |
| JPEG20 | 27.733 | 24.274 |
| JPEG25 | 27.799 | 24.273 |
| JPEG30 | 27.774 | 24.270 |
| JPEG35 | 27.812 | 24.279 |
| JPEG40 | 27.897 | 24.288 |
| JPEG50 | 27.878 | 24.269 |
| JPEG60 | 27.876 | 24.262 |
| JPEG70 | 27.826 | 24.239 |
| JPEG80 | 27.824 | 24.215 |
| JPEG90 | 27.820 | 24.250 |
| JPEG100 | 27.907 | 24.338 |

### 5.1.4. Crop attack

The cropped image is a one-off type in the geometric attack. Crop working is applied to the image such as is right to left, top to bottom, square, circle, rectangular type crop of the image. fig. 9(a) shows that $2^{nd}$ watermarked image after crop 75 attack, fig. 9(b) shows that $1^{st}$ watermarked image after crop 75 attack, fig. 9(c) is retrieved watermark image $1^{st}$ after crop 75 attack, fig. 9(d) is retrieved watermark image $2^{nd}$ after crop 75 attack, fig. 9(e) is retrieved watermark image $3^{rd}$ after crop 75 attack by using “Stir Mark 4.0”. Table-4(a) shows PSNR value after crop 75 attacks is using the $1^{st}$ watermarked image; Table-4(b) shows PSNR value after crop 75 attack using $2^{nd}$ watermarked images.



(a) (b) (c)

(d) (e)

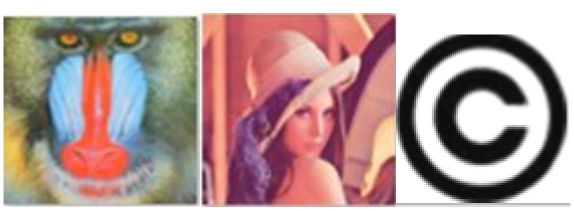
Figure 9 attack of after crop75; (a) 2nd watermarked image, (b)1st watermarked image, (c) retrieved watermark image 1st, (d) retrieved watermark image 2nd, (e) retrieved watermark image3rd.

Table4: (a) for PSNR value of watermarked image 1st & (b) for PSNR value of watermarked image 2nd for Crop attack.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| Crop1 | 20.1465 | 13.402 |
| Crop2 | 17.799 | 15.755 |
| Crop5 | 17.055 | 17.092 |
| Crop10 | 16.147 | 16.990 |
| Crop15 | 17.883 | 16.281 |
| Crop20 | 15.173 | 16.232 |
| Crop25 | 15.116 | 15.510 |
| Crop50 | 16.192 | 15.530 |
| Crop75 | 17.767 | 17.023 |

### 5.1.5. Median Filter attack

Median filter smoothness the image by utilizing the median of the neighborhood. The median filter is smoothening additive white noise. Fig. 10(a) shows that 2nd watermarked image after median cut 9 attacks, fig. 10(b) 1st shows that watermarked image after crop 75 attack, fig. 10(c) is retrieved watermark image 1staftermedian cut 9 attacks, fig. 10(d) is retrieved watermark image 2nd after median cut 9 attacks, fig. 10(e) is retrieved watermark image 3rd after median cut 9 attacks by using "Stir Mark 4.0". Table-5(a) shows PSNR value after median cut 9 attacks using the 1st watermarked image, Table-5(b) shows PSNR value after median cut 9 attack using 2nd watermarked images.



(a) (b) (c)

(d) (e)

Figure 10 attack of after median cut 9; (a) 2nd watermarked image, (b)1st watermarked image, (c) retrieved watermark image 1st, (d) retrieved watermark image 2nd, (e) retrieved watermark image 3rd.

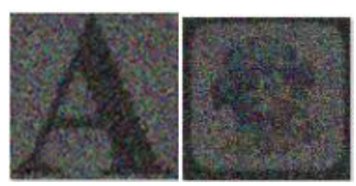
Table 5: (a) for PSNR value of watermarked image 1st & (b) for PSNR value of watermarked image 2nd for Median attack.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| Median3 | 27.495 | 24.795 |
| Median5 | 28.3324 | 24.331 |
| Median7 | 28.3323 | 24.168 |
| Median9 | 28.1467 | 24.024 |

### 5.1.6. Add noise attack

Salt pepper noise added into the watermarked image. Fig. 11(a) shows that 2nd watermarked image after add noise 80 attack, fig. 11(b) shows that 1st watermarked image after add noise 80 attack, fig. 11(c) is retrieved watermark image 1st after add noise 80 attack, fig. 11(d) is retrieved watermark image 2nd after add noise 80 attack, fig. 11(e) is retrieved watermark image 3rd after add noise 80 attack by using "Stir Mark 4.0". Table 6(a) shows PSNR value after adds noise 80 attacks using the 1st watermarked image; Table-6(b) shows PSNR value after adds noise 80 attack using 2nd watermarked images.

(a) (b) (c)



(d) (e)

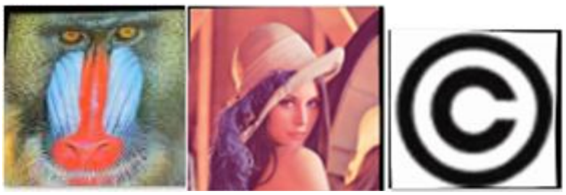
Figure 11 attack of after add noise 80; (a) 2nd watermarked image, (b)1st watermarked image, (c) retrieved watermark image 1st, (d) retrieved watermark image 2nd, (e) retrieved watermark image3rd.

Table 6: (a) for PSNR value of watermarked image 1st & (b) for PSNR value of watermarked image 2nd for add noise attack.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| Noise 0 | 14.875 | 24.341 |
| Noise 20 | 27.900 | 13.797 |
| Noise 40 | 13.425 | 13.195 |
| Noise 60 | 12.965 | 12.705 |
| Noise 80 | 12.747 | 12.453 |
| Noise 100 | 12.626 | 12.342 |

### 5.1.7. Random distortion attack (RND dist.)

Attack image and Watermarked image is used RND dist. fig. 12(a) shows that 2nd watermarked image after RND dist. 0.95 attacks, fig. 12(b) shows that 1st watermarked image after RND dist. 0.95 attacks, fig. 12(c) is retrieved watermark image $1^{st}$ after RND dist. 0.95 attacks, fig. 12(d) is retrieved watermark image $2^{nd}$ after RND dist. 0.95 attacks, fig. 12(e) is retrieved watermark image $3^{rd}$ after RND dist. 0.95 are used "Stir Mark 4.0". Table-7(a) shows PSNR value after RND dist. 0.95 attacks using a $1^{st}$ watermarked image, Table-7(b) shows PSNR value after RND dist. 0.95 attack using $2^{nd}$ watermarked images.



(a) (b) (c)

(d) (e)

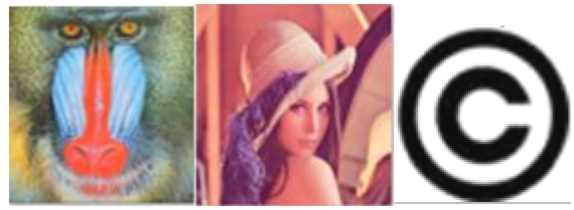
Figure 12 attack of after random distortion 0.95; (a) $2^{nd}$ watermarked image, (b)$1^{st}$ watermarked image, (c)retrieved watermark image $1^{st}$, (d)retrieved watermark image $2^{nd}$, (e)retrieved watermark image $3^{rd}$.

Table 7: (a) for PSNR value of watermarked image $1^{st}$ & (b) for PSNR value of watermarked image $2^{nd}$ for random distortion attack.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| Rnd dist. 0.95 | 22.616 | 17.038 |
| Rnd dist. 1.1 | 22.227 | 16.715 |
| Rnd dist. 1.05 | 22.360 | 16.839 |
| Rnd dist. 1 | 22.466 | 16.938 |

### 5.1.8. Rescale attack

Rescale is one type of geometric attacks. Scaling transformation alters the size of an image. Fig. 13(a) shows that $2^{nd}$ watermarked image after Rescale 110 attack, fig. 13(b) shows that $1^{st}$ watermarked image after Rescale 110 attack, fig. 13(c) is retrieved watermark image $1^{st}$ after Rescale 110 attack, fig. 13(d) is retrieved watermark image $2^{nd}$ after Rescale 110 attack, fig. 13(e) is retrieved watermark image $3^{rd}$ after Rescale 110 attacks by using "Stir Mark 4.0". Table-8(a) shows PSNR value after Rescale attack using a $1^{st}$ watermarked image, Table-8(b) shows PSNR value after Rescale attack using $2^{nd}$ watermarked images.



(a) (b) (c)

(d) (e)

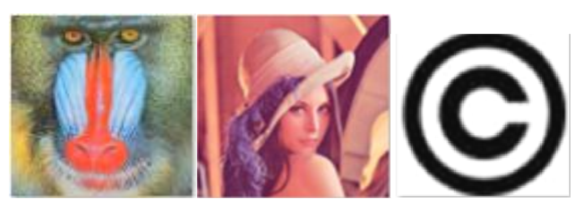
Figure 13 attack of after Rescale 110; (a) $2^{nd}$ watermarked image, (b)$1^{st}$ watermarked image, (c) retrieved watermark image $1^{st}$, (d) retrieved watermark image $2^{nd}$, (e) retrieved watermark image $3^{rd}$.

Table 8: (a) for PSNR value of watermarked image $1^{st}$ & (b) for PSNR value of watermarked image $2^{nd}$ for rescale attacks.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| Rescale 50 | 27.875 | 24.281 |
| Rescale 75 | 28.0146 | 24.405 |
| Rescale 90 | 27.465 | 25.577 |
| Rescale 110 | 28.051 | 25.529 |
| Rescale 150 | 27.097 | 25.783 |
| Rescale 200 | 27.907 | 25.940 |

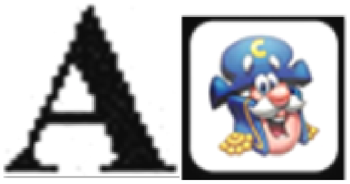
### 5.1.9. RML (Removing of lines) attack

RML is an attack on some copyright marking system. Benchmark "Stir Mark 4.0" is used finding the result. Fig. 14(a) is $2^{nd}$ watermarked image after RML 60 attack, fig. 14(b) is $1^{st}$ watermarked image after RML 60 attack, fig. 14(c) is retrieved watermark image $1^{st}$ after RML 60 attack, fig. 14(d) is retrieved watermark image $2^{nd}$ after RML 60 attack, fig. 14(e) is retrieved watermark image $3^{rd}$ after RML 60 attacks by using "Stir Mark 4.0". Table-9(a) shows PSNR value after RML attack using a $1^{st}$ watermarked image, Table-9(b) shows PSNR value after RML attack using $2^{nd}$ watermarked images.



(a) (b) (c)



(d) (e)

Figure 14 attack of after RML 60; (a) 2nd watermarked image, (b)1st watermarked image, (c) retrieved watermark image 1st, (d) retrieved watermark image 2nd, (e) retrieved watermark image3rd.

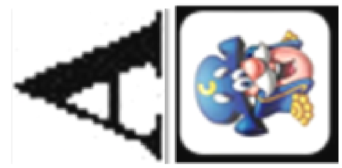
Table 9: (a) for PSNR value of watermarked image 1st & (b) for PSNR value of watermarked image 2nd of remove of line attacks.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| RML 10 | 28.296 | 24.462 |
| RML 20 | 28.538 | 23.651 |
| RML 30 | 28.276 | 24.751 |
| RML 40 | 28.737 | 23.239 |
| RML 50 | 28.465 | 24.342 |
| RML 60 | 28.569 | 23.769 |
| RML 70 | 28.433 | 24.213 |
| RML 80 | 28.478 | 23.957 |
| RML 90 | 28.712 | 23.503 |
| RML 100 | 28.242 | 24.578 |

### 5.1.10. Rotation (ROT), Rotation with cropping (ROTCROP) and scaling (ROTSCALE) attack

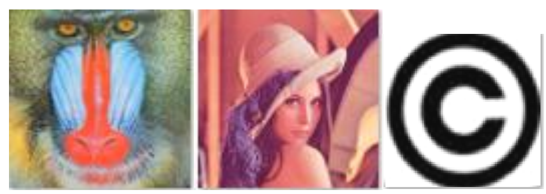
Rotate is applied to an image by repositioning it along a circular path in the x-y plane. Rotation through an angle Ø about origin clockwise. Rotate image is moving left, right, top, bottom. Fig. 15(a) shows that 2nd watermarked image after Rotation 90◦ attack, fig. 15(b) is 1st watermarked image after Rotation 90◦ attack, fig. 15(c) is retrieved watermark image 1st after Rotation 90◦ attack, fig.15(d) is retrieved watermark image 2nd after Rotation 90◦ attack, fig. 15(e) is retrieved watermark image 3rd after Rotation 90◦ attack by using "Stir Mark 4.0". Fig. 16(a) 2nd watermarked image after Rotation with crop 2 attack, fig. 16(b) 1st watermarked image after Rotation with crop 2 attack, fig.16(c) is retrieved watermark image 1st after Rotation with crop 2 attack, fig. 16(d) is retrieved watermark image 2nd after Rotation with crop 2 attack, fig. 16(e) is retrieved watermark image 3rd after Rotation with crop 2 attack by using "Stir Mark 4.0". Fig. 17(a) is 2nd watermarked image after Rotation with scale -0.5 attack, fig. 17(b) is 1st watermarked image after Rotation with scale -0.5 attack, fig. 17(c) is retrieved watermark image 1st after Rotation with scale -0.5 attack, fig. 17(d) is retrieved watermark image 2nd after Rotation with scale -0.5 attack, fig.17(e) is retrieved watermark image3rd after Rot scale -0.5 attack by using "Stir Mark 4.0". Table-10(a) shows PSNR value after Rotation attack using 1st watermarked image, Table-10(b) shows PSNR value after Rotation attack using 2nd watermarked images.

(a) (b) (c)



(d) (e)

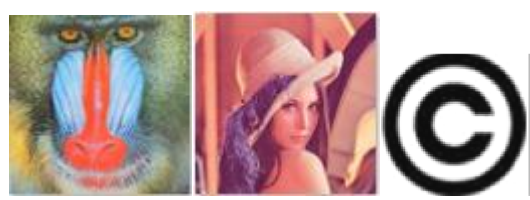
Figure 15 attack of after ROT 90◦ ; (a) 2nd watermarked image, (b) 1st watermarked image, (c) retrieved watermark image 1st, (d) retrieved watermark image 2nd, (e) retrieved watermark image 3rd.



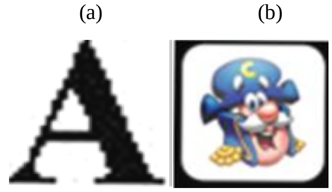
(a) (b) (c)

(d) (e)

Figure 16 attack of after ROTCROP 2; (a) 2nd watermarked image (b) 1st watermarked image, (c)retrieved watermark image 1st, (d)retrieved watermark image 2nd, (e)retrieved watermark image 3rd.



(a) (b) (c)



(d) (e)

Figure 17 attack of after ROTSCALE -0.5; (a)2nd watermarked image, (b)1st watermarked image, (c) retrieved watermark image 1st, (d)retrieved watermark image 2nd, (e) retrieved watermark image 3rd.

Table 10: (a) for PSNR value of watermarked image 1st & (b) for PSNR value of watermarked image 2nd for rotation attacks.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| ROT 0.5 | 23.228 | 21.395 |
| ROT -0.5 | 24.982 | 21.383 |
| ROT 1 | 23.043 | 20.196 |
| ROT -1 | 23.368 | 20.247 |
| ROT 2 | 21.008 | 18.732 |
| ROT -2 | 21.335 | 18.470 |
| ROT 5 | 14.419 | 16.645 |
| ROT 10 | 16.627 | 15.015 |
| ROT 15 | 15.857 | 14.232 |
| ROT 30 | 14.819 | 13.113 |
| ROT 45 | 14.552 | 12.740 |
| ROT 90 | 15.560 | 16.780 |
| ROT Crop 0.5 | 26.800 | 22.199 |
| ROT Crop -0.5 | 26.651 | 22.234 |
| ROT Crop 0.25 | 27.624 | 23.576 |
| ROT Crop -0.25 | 28.188 | 23.430 |
| ROT CROP 0.75 | 25.820 | 21.806 |
| ROT Crop -0.75 | 25.708 | 21.810 |
| ROT Crop 1 | 25.570 | 21.580 |
| ROT Crop -1 | 24.928 | 21.579 |
| ROT Crop 2 | 23.736 | 20.735 |
| ROT Crop -2 | 23.042 | 20.781 |
| ROT Scale 0.5 | 26.722 | 22.188 |
| ROT Scale -0.5 | 26.595 | 22.190 |
| ROT Scale 0.25 | 27.711 | 23.766 |
| ROT Scale -0.25 | 28.100 | 23.687 |
| ROT Scale 0.75 | 26.270 | 21.787 |
| ROT Scale -0.75 | 25.581 | 21.795 |
| ROT Scale 1 | 25.481 | 21.550 |
| ROT Scale -1 | 24.797 | 21.555 |
| ROT Scale 2 | 23.027 | 20.724 |
| ROT Scale -2 | 23.796 | 20.750 |

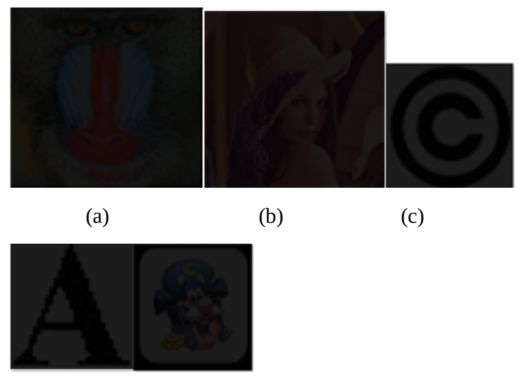
### 5.1.11. Convolution filter

Convolution filter 1 and convolution filter 2 are used "Stir Mark 4.0" allotted and attack of image. The working use n×m matrix is assign to the image. Fig. 18(a) is 2nd watermarked image after convolution filter1 attack, fig. 18(b) is 1st watermarked image after convolution filter1 attack, fig. 18(c) is retrieved watermark image 1st after convolution filter1 attack, fig. 18(d) is retrieved watermark image 2nd after convolution filter1 attack, fig. 18(e) is retrieve watermark image 3rd after convolution filter1 attack, Fig.19(a) is 2nd watermarked image after convolution filter 2 attack, fig. 19(b) is 1st watermarked image after convolution filter 2 attack, fig. 19(c) is retrieved watermark image 1st after convolution filter 2 attack, fig. 19(d) is retrieved watermark image 2nd after convolution filter 2 attack, fig. 19(e) is retrieved watermark image3rd after convolution filter2 attack.Table-11(a) shows PSNR value after convolution attack using the 1st watermarked image, Table-11(b) shows PSNR value after convolutions attack using 2nd watermarked images.

(a) (b) (c)

(d) (e)

Figure18attack of after convolution1 (a) 2ndwatermarked image, (b) 1stwatermarked image, (c) retrieved watermark image 1st, (d) retrieved watermark image 2nd, (e)retrieved watermark image3rd.



(a) (b) (c)

(d) (e)

Figure 19 attack of after convolution 2; (a) 2nd watermarked image, (b) 1stwatermarked image, (c) retrieved watermark image 1st, (d) retrieved watermark image 2nd, (e) retrieved watermark image3rd.

Table 9: (a) for PSNR value of watermarked image 1st & (b) for PSNR value of watermarked image 2nd for convolution attacks.

| Attacks | (a)PSNR | (b)PSNR |
|---|---|---|
| Convolution 1 | 15.779 | 13.795 |
| Convolution 2 | 13.149 | 11.834 |

## 5.2. ROBUSTNESS EVALUATION

Image retrieve is to match embedded information and the extracted watermark information. They use measurement as

### 5.2.1. Normalized correlation coefficient (NCC)

NCC has been used for quantitative evaluation between the extracted watermark and different type attacks. NCC signal between range value of (-1 to 1). NCC value stated in equation (4), is the between original watermark and the retrieved watermark image 1st, 2nd, and 3rd after attacks, as shown in table12.

$$NCC = \frac{\sum_{I=1}^{M}\sum_{j=1}^{N} R(i,j).\bar{R}(i,j)}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N} R(i,j)^2 \sum_{i=1}^{M}\sum_{j=1}^{N} R'(i,j)^2}} \quad (4)$$

Where, $\bar{R}(i,j)$ is the extracted watermark, R (i, j) is the original watermark, M and N denoted row and column size.

Table12: For NCC value of different attacks using retrieved watermark image.

| Attacks | Retrieved image 1st | Retrieved image 2nd | Retrieved image 3rd |
|---|---|---|---|
| Affine 1 | 0.317 | 0.996 | 0.988 |
| Affine 2 | 0.317 | 0.996 | 0.987 |
| Affine 3 | 0.316 | 0.996 | 0.989 |
| Affine 4 | 0.317 | 0.995 | 0.971 |
| Affine 5 | 0.315 | 0.994 | 0.988 |
| Affine 6 | 0.316 | 0.993 | 0.988 |
| Affine 7 | 0.314 | 0.996 | 0.986 |
| Affine 8 | 0.317 | 0.994 | 0.988 |
| Noise 0 | 0.223 | 0.509 | 0.501 |
| Noise 20 | 0.31764 | 0.635 | 0.627 |
| Noise 40 | 0.31765 | 0.635 | 0.627 |
| Noise 60 | 0.3176 | 0.937 | 0.929 |
| Noise 80 | 0.31764 | 0.996 | 0.988 |
| Noise 100 | 0.3175 | 0.925 | 0.917 |
| Rnd dist. 0.95 | 0.000 | 0.996 | 0.988 |
| Rnd dist. 1.1 | 0.000 | 0.996 | 0.988 |
| Rnd dist. 1.05 | 0.000 | 0.992 | 0.984 |
| Rnd dist. 1 | 0.000 | 0.996 | 0.988 |
| Rot Crop 0.5 | 0.290 | 0.662 | 0.654 |
| Rot Crop -0.5 | 0.00 | 0.686 | 0.678 |
| Rot Crop 0.25 | 0.282 | 0.584 | 0.576 |
| Rot Crop -0.25 | 0.015 | 0.549 | 0.541 |
| Rot Crop 0.75 | 0.282 | 0.690 | 0.682 |
| Rot Crop -0.75 | 0.313 | 0.682 | 0.674 |
| Rot Crop 1 | 0.290 | 0.752 | 0.745 |
| Rot Crop -1 | 0.070 | 0.749 | 0.741 |
| Rot Crop 2 | 0.294 | 0.870 | 0.862 |
| Rot Crop -2 | 0.000 | 0.725 | 0.717 |
| JPEG 20 | 0.207 | 0.533 | 0.525 |
| JPEG 30 | 0.206 | 0.501 | 0.494 |
| JPEG 40 | 0.227 | 0.505 | 0.498 |
| JPEG 50 | 0.223 | 0.513 | 0.505 |
| JPEG 60 | 0.227 | 0.509 | 0.501 |
| JPEG 70 | 0.223 | 0.501 | 0.494 |
| Median 3 | 0.00784 | 0.525 | 0.517 |
| Median 5 | 0.000 | 0.529 | 0.521 |
| Median 7 | 0.000 | 0.541 | 0.533 |
| Median 9 | 0.000 | 0.552 | 0.545 |
| Convolution 1 | 0.152 | 0.533 | 0.525 |
| Convolution 2 | 0.305 | 0.823 | 0.815 |

### 5.2.2. Bit Error Rate (BER)

BER is evaluating the robustness of watermarking. ⊕ Means X-or operation. BER is the number of bit error per unit time. BER is used in between extract watermark and different type attack. BER value stated in equation (5), is the between original watermark and the retrieved watermark image 1st, 2nd, and 3rd after attacks, as shown in table13.

$$BER = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} R(i,j) \oplus \bar{R}(i,j)}{M \times N} \quad (5)$$

Where, $\bar{R}$(i, j) is the extracted watermark and R (i, j) is the original watermark.M and N denoted row and column size of extracted watermark.

Table 13: For BER value of different attacks using retrieved watermark image.

| Attacks | Retrieved image 1st | Retrieved image 3rd | Retrieved image 4th |
|---|---|---|---|
| Affine 1 | 0.000005 | 0.00001 | 0.000646 |
| Affine 2 | 0.000029 | 0.00008 | 0.000622 |
| Affine 3 | 0.000437 | 0.00132 | 0.000210 |
| Affine 4 | 0.000607 | 0.00182 | 0.000045 |
| Affine 5 | 0.000440 | 0.00133 | 0.000207 |
| Affine 6 | 0.000482 | 0.00144 | 0.000169 |
| Affine 7 | 0.000434 | 0.00130 | 0.000216 |
| Affine 8 | 0.000375 | 0.00113 | 0.00275 |
| Noise 0 | 0.00000 | 0.0000 | 0.000651 |
| Noise 20 | 0.00037 | 0.00084 | 0.000617 |
| Noise 40 | 0.000122 | 0.00053 | 0.000516 |
| Noise 60 | 0.000186 | 0.000229 | 0.000443 |
| Noise 80 | 0.000234 | 0.000114 | 0.000417 |
| Noise 100 | 0.000243 | 0.000202 | 0.000379 |
| Rnd dist. 0.95 | 0.000383 | 0.00116 | 0.0000 |
| Rnd dist. 1.1 | 0.000389 | 0.00117 | 0.0000 |
| Rnd dist. 1.05 | 0.000387 | 0.00117 | 0.000387 |
| Rnd dist. 1 | 0.000385 | 0.00120 | 0.000385 |
| Rotation 0.5 | 0.000343 | 0.001053 | 0.000301 |
| Rotation -0.5 | 0.000412 | 0.001259 | 0.000233 |
| Rotation 1 | 0.000458 | 0.001377 | 0.000192 |
| Rotation 2 | 0.000557 | 0.001671 | 0.000131 |
| Rotation 5 | 0.000618 | 0.001854 | 0.000033 |
| Rotation 15 | 0.000640 | 0.001919 | 0.000011 |
| Rotation 30 | 0.000643 | 0.001930 | 0.000010 |
| Rotation 45 | 0.000646 | 0.001934 | 0.00006 |
| Rotation 90 | 0.000 | 0.000 | 0.000651 |
| Strd dist. 0.95 | 0.000212 | 0.000446 | 0.000416 |
| Strd dist. 1.1 | 0.000213 | 0.000458 | 0.000413 |
| Strd dist. 1.05 | 0.000212 | 0.000420 | 0.000415 |
| Strd dist. 1 | 0.000214 | 0.000439 | 0.000416 |

## 6. Conclusion

In this paper, a robust and imperceptible digital image watermarking method is generated by using SVD-DWT-DCT. To increase robustness of this proposed method, three watermark images are hided in fourth another watermark image, which is having the same size of cover image. This forth watermark image then hided in the cover image. On this watermarked image different types of attacks are applied using Benchmark, "Stir Mark 4.0". The Imperceptibility tested by PSNR value. It can be concluded that the fidelity of the watermarked image is good with the method stated in this paper. Then NCC and BER between retrieved watermark and original watermark is estimated. The estimated result shows better Robustness. From this, it is observed that the retrieved watermark is highly efficient with the proposed method. Here it can be concluded that the algorithm in this paper has two advantages, firstly good in PSNR values (no degradation of image) and secondly higher NCC values (improved robustness of watermark).

## REFERENCE


[1]. Nazir A. Loani, Nasir N. Hurrah, Shabir A. Parah, Jong Weon Lee, Javid A. Sheikh and G. Mohiuddun Bhat "Secure and Robust Digital Image Watermarking Using Coefficient Differencing and Chaotic Encryption" IEEE access, vol. 6, pp.19876-19897, 2018.
[2]. Nazir A. Loani, Nasir N. Hurrah, Shabir A. Parah, Jong Weon Lee, Javid A. Sheikh and G. Mohiuddun Bhat "Secure and Robust Digital Image Watermarking Using Coefficient Differencing and Chaotic Encryption" IEEE access, vol. 6, pp.19876-19897, 2018.
[3]. Ferda Ernawan and Muhammad Nomani Kabiara "Robust Image Watermarking Technique with an Optimal DCT-Psych visual Threshold" IEEE, 2018.
[4]. Patrick Schuch, Simon Schulz and Christoph Busch "Survey on the impact of fingerprint image enhancement" IET Biome, Vol. 7 Iss. 2, pp. 102-115, 2018.
[5]. Yahya AL-Nabhani, Hamid A. Jalab, Ainuddin Wahid and Rafidah Md Noor "Robust watermarking algorithm for digital images using discrete wavelet and probabilistic neural network" computer and information science, vol. 27, pp.393-401, 2015.
[6]. Amit Phadikar, Santi P. Maity, Bhupendra Verma "Region-based QIM digital watermarking scheme for image data base in DCT domain" computer an electric engineering, vol. 37, pp.339-355, 2011.
[7]. https://www.cl.cam.ac.uk/fapp2/watermarking/stirmark.


## AUTHOR'S PROFILE

**Sushma Kejgir** is an assistant professor of Department of Electronics and Tele-Communication engineering at Shri Guru Gobind Singhji Institute of Engineering and Technology, Vishnupuri, Nanded, India. Her subject interest includes Digital Image Watermarking and Electromagnetic engineering. He is a review of IEEE Transaction on system man cybernetics part B and Elsevier journal computer interface and standard.

**Foram Suratwala** is an MTech student of Department of Electronics and Tel-Communication engineering at Shri Guru Gobind singhji Institute of Engineering and Technology, Vishnupuri, Nanded, India. Her research interests include image processing and digital watermarking.