# A Hybrid Wavelet-Shearlet Approach to Robust Digital Image Watermarking

Ahmad B. A. Hassanat[†], V. B. Surya Prasath[*], Khalil I. Mseidein[†], Mouhammd Al-awadi[†] and Awni Mansoar Hammouri[†]
[†]Department of Information Technology, Mutah University, Karak, Jordan
[*]Computational Imaging and VisAnalysis (CIVA) Lab, Department of Computer Science, University of Missouri-Columbia, USA
E-mail: prasaths@missouri.edu

**Keywords:** watermarking, shearlet transform, Arnold transform, discrete wavelet transform

**Received:** December 6, 2016

*Watermarking systems are one of the most important techniques used to protect digital content. The main challenge facing most of these techniques is to hide and recover the message without losing much of information when a specific attack occurs. This paper proposes a novel method with a stable outcome under any type of attack. The proposed method is a hybrid approach of three different transforms, discrete wavelet transform (DWT), discrete shearlet transform (DST) and Arnold transform. We call this new hybrid method SWA (shearlet, wavelet, and Arnold). Initially, DWT applied to the cover image to get four sub-bands, we selected the HL (High-Low) sub band of DWT, since HL sub-band contains vertical features of the host image, where these features help maintain the embedded image with more stability. Next, we apply the DST with HL sub-band, at the same time applying Arnold transform to the message image. Finally, the output that obtained from Arnold transform will be stored within the Shearlet output. To evaluate the proposed method we used six performance evaluation measures, namely, peak signal to noise ratio (PSNR), mean squared error (MSE), root mean squared error (RMSE), signal to noise ratio (SNR), mean absolute error (MAE) and structural similarity (SSIM). We apply seven different types of attacks on test images, as well as apply combined multi-attacks on the same image. Extensive experimental results are undertaken to highlight the advantage of our approach with other transform based watermarking methods from the literature. Quantitative results indicate that the proposed SWA method performs significantly better than other transform based state-of-the-art watermarking approaches.*

*Povzetek: Opisana je robustna metoda digitalnega vodnega tiska, tj. vnosa kode v sliko.*

## 1 Introduction

With the exponential growth of digital contents in the internet, there is a strong need to protect these contents with more security automatically. This open online environment needs more efficient techniques to save original content creators, and author's rights. Digital watermarking systems can significantly contribute to protect information and files. Generally, people need an easy-to-use model to protect their files, texts, and images. The availability of automatic tools that provide such services for documents are currently restricted and difficult to use. Although, there are many previous works focused on certain solutions to solve this security problem, we certainly need more research to improve the efficiency of these existing methods. This paper provides a new hybrid approach based on some efficient transforms and provides an overview of the relevant issues and definitions related to digital image watermarking. Copyright nowadays is one of the most important research areas; digital watermarking is considered as one of the important automatic signal/image processing technique that enables us to hide our information behind a noisy signal. This signal may be image, audio or video etc.

An important extension of watermarking area is the image watermarking; here we embedded a watermark image within a cover image. The produced watermarked image is a combination of cover image and the watermark. The watermark is the information to be embedded, on the other hand the host or cover image is the signal where the watermark is embedded. In general, Watermarks can be classified depending on several criteria such as: domain that can be applied by the watermark, type the watermark used visibility of watermark to user and the application used to create the watermark. Figure 1 shows these broad classifications.

### 1.1 The usage of image watermarking

The growth of digital computing technology in recent times was the reason beyond the widespread use of digital media such as video digital, documents and digital images. As a result of the increase in speed of transmission and distribution it is easy to obtain digital content. Despite the abundance of digital products, however this technology lacks protection because of illegal use, and imitations. The protection of intellectual property rights for digital media has been the attention the focus of many researchers in the past. Using a digital watermark technology is a suc-

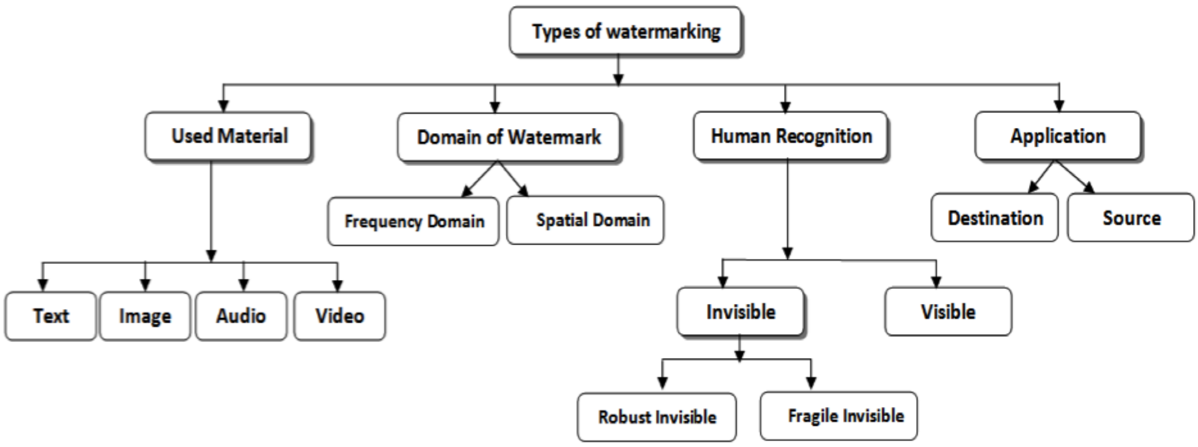


Figure 1: Main types of Watermarking techniques.

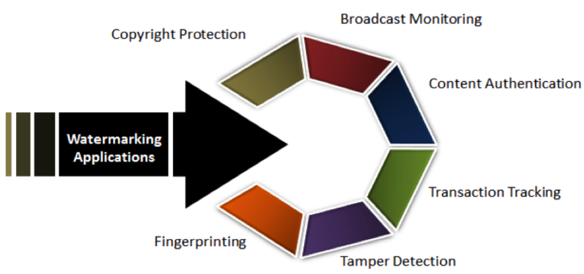


Figure 2: Application areas of watermarking techniques.

cessful choice to solve the digital content protection problem. There are, in general, six types of watermarking applications are presented; copyright protection, fingerprinting, broadcast monitoring, content authentication, transaction tracking, and tamper detection, these applications are mentioned in Figure 2.

Recently, the advance of editing capabilities and the wide availability due to internet penetration in the world, digital forgery has become easy, and difficult to prevent in general. Therefore, there is a need to protect end-user privacy, security, and copyright issues for content generators [13]. The application areas include biometrics, medical imagery, telemedicine, etc [4, 38, 39, 40, 41, 42].

Nowadays, digital watermarking is used mainly for the protection of copyrights. Moreover, it was used early to send sensitive confidential information across the communicative signal channels. Thus, applying watermark has been occupying the attention of researchers in the last few decades. As a consequence of tremendous growth of computer networks and data transfer over the web; huge amounts of multimedia data are vulnerable to unauthorized access, for e.g., web images which can easily be duplicated and distributed without eligibility. Image watermarking provides copyright protection for documents and multimedia in order to protect intellectual property and data security.

## 1.2 Contributions

In this work we will provide an overview five different transform based methods one of which is a new proposed hybrid transform approach. We considered the transform of an image using discrete cosine transform (DCT), discrete wavelet transform (DWT), DWT with DCT, and DWT with Arnold transform in addition to our proposed hybrid method which combines discrete shearlet transform (DST) with these previous transforms. Also, we considered adding attacks like crop image, salt and pepper noise, Gaussian noise, and etc. The schemes we are providing here are resilient to these types of attacks and novel in this scenario. In this paper, we will study in details the common methods that used to protect the original image after adding attacks. This work also presents a comprehensive experimental study of these transform based watermarking algorithms. Another important contribution of this study is a hybrid method which is based on a fusion of different transforms taking advantages of each transforms discriminability. Initially, we will design our new method; then we test it and compare our results with the results of the previous transform and combination of them. To ensure the efficiency of our hybrid, we will use different performance measures to quantitatively benchmark it on various test images and attacks.

In the digital image watermarking area, efficiency of any proposed algorithm can be evaluated based on a host of image and embedded image (message). One of the most common image is the "Lena" which is traditionally used as a host. Also, the copyright image has been used widely as a message. Apart from these images, there are other standard test images such as Cameraman, Baboon, Peppers, Airplane, Boat, Barbara, Elaine, Man, Bird, Couple, House, and Home which are widely used in benchmarking evaluations. With the development of a new watermarking technology, it is necessary to protect the images against multiple types of attack, such as compression, Gaussian filter, pepper and salt noise, median filter, cropping, resize, and rotation. The Watermark is said to be robust against some attack if we can detect the message after that particular attack. In our proposed approach, to ensure and reduce influence of watermarked image in any attack, shearlet transform is applied at the HL (high-low) sub-band, where this sub-band retains features of original image. shearlets generate a series of matrices that obtained from the HL subband. In this way, shearlets applied to specific pixels of the original image with these features represented in 61 different matrices. As a consequence, any applied attack is distributed to all of these matrices, and the probability of changing the value of the pixel that has been embedding by the attack is low. Based on this observation, our proposed hybrid SWA approach's results were stable or semi-static whatever the type of attack and its value. Finally, the Arnold transform was applied to the message image to achieve the best protection against the unauthorized change in pixel values.

We organized the rest of the paper as follows. Section 2 provides a brief overview of literature on watermarking with special emphasis on transform domain watermarking techniques. Section 3 provides a detail explanation of the proposed hybrid shearlet, wavelet, and Arnold (SWA) method. Section 4 provides detailed experimental results and discussions with Section 5 concluding the paper.

# 2 Literature review

Recently, research activities in image watermarking area has seen a lot of progress, and become more specialized. Some researchers were focusing on improving the properties of the watermark or applications, while others were focusing on improving the efficiency of the techniques used to embed, and extract the watermarked image, taking into consideration attacks. Watermarking techniques can be classified generally in to two broad domains; spatial domain, and transform domain.

## 2.1 Spatial domain watermarking

Classical techniques do watermarking in the spatial domain, where the message image is included by adjusting the pixel values of the original (host) image . Least Significant bit (LSB) method is considered one of the most important and most common examples of the use of the spatial domain techniques for the watermark. There are two main procedures to any watermarking technique model; embedding procedure and extraction procedure. For technique of LSB in embedding phase, the host image (original or cover) and the message image (watermark) being read, both images must be gray. However, this method suffers from the problem of an impaired ability to hide information [20], therefore it is easy to extract the image hidden within the original image by unauthorized persons, moreover, the quality of watermarking is not good when embedding procedure, and extraction procedure are combined. In other words, the results achieved by spatial domain methods are not good enough, especially when the intensity of the pixel changed directly, which affects the value of this pixel.

Since typical spatial domain methods suffer from much vulnerability, many authors have tried to improve these. For example, [50] introduced two different methods to improve the technique of LSB, in the first method LSB substituted with a pseudo-noise (PN) sequence, and a second method that adds both together. However, this improvement also gives unsatisfactory results after adding any type of noise.

## 2.2 Transform domain watermarking

These techniques rely on hiding images in the transformed coefficients, thereby giving further information that helps secreting against any attack. As a consequence, majority of recent studies in the field of digital image watermarking use the frequency domain. Use of the frequency domain gives results better than the spatial domain in terms of robustness [28] . There are many transforms can be used for images watermarking based on frequency domain (FD), such as continuous wavelet transform (CWT), discrete cosine transform (DCT), short time Fourier transform (STFT), discrete wavelet transforms (DWT), Fourier transform, and combinations of DCT and DWT. We very briefly review these well-known transforms as they are relevant to the hybrid method proposed here (Section 3).

### 2.2.1 Discrete cosine transform (DCT)

One of the most important methods that based on the frequency domain is the discrete cosine transform (DCT). Typically in DCT based approaches, the image is represented in the form of a set of sinusoids with changeable magnitudes and frequencies, where the image is split into three different sections of frequencies; low frequency (LF), medium frequency (MF) and high frequency (HF). Data or message will be hidden in the medium frequency region; since it is considered to be the best place, whereas if the message is stored in low frequency regions it will be visible to the naked human eyes. Thus, for the areas of higher frequency, if the message is stored in this region, the resulting image will be distorted because this frequency spreads the biggest place of the block on the bottom right corner. Consequently, this will cause local deformation combined with the edges, thus the places where the areas are of the medium frequency, does not affect the quality of the image. The DCT is utilized in a number of earlier studies, see for e.g. [49] who proposed a new model based DCT technique within a specific scheme for the watermark, in which DCT increased the resistance against attacks, mainly JPEG compression attacks.

#### 2.2.2 Discrete wavelet transform (DWT)

Discrete wavelet transform (DWT) is based on wavelets and is sampled discretely. The goal of using DWT is to convert an image from the spatial domain to the frequency domain in a locality preserving way. In DWT transform, coefficients separate the high and low frequency information in an image on a pixel-to-pixel basis. Original signal is divided into four mini signals - low-low (LL), low-high (LH), high-low (HL), and high-high(HH), these wavelets are generated from the original signal by dilations and translations. It is commonly used in various image processing applications, and in particular in the application of watermarking [48]. DWT can find appropriate area to embed the message efficiently, this means that the message will be invisible to the naked human eyes.

#### 2.2.3 Joint transforms (DCT and DWT)

According to the advantages of the last two methods, namely DCT and DWT, we notice that each one is been characterized by certain positive aspects, however there are limitations that restrict their application efficiency. To improve the performances, several studies combined these two transforms based techniques, see for e.g. [8, 2, 5]. DCT achieves high results in the robust of hiding data, but it produces a distorted image, DWT produces high-quality image, but it achieves bad results with the addition of the attack. One of the proposed solutions to resolve this issue is a hybrid method combined two techniques; this hybrid method usually called joint DWT-DCT, the joint method is common use in signal processing application.

# 3 Proposed hybrid approach

The proposed approach is based on a hybrid approach combining three different transforms namely: discrete wavelet transform (DWT), discrete shearlet transform (DST) and Arnold transform. We named this a new hybrid model SWA, this abbreviation comes from the name of transforms utilized here; Shearlet, Wavelet, and Arnold transforms respectively. In this section, we will explain the salient points of these three transforms as well as our method of merging them together for the purposes of digital image watermarking.

## 3.1 Discrete wavelet transform (DWT)

As mentioned in Section 2.2.2, DWT is perhaps one of the most commonly used transform in the field of watermarking, where it is most widely used in image and videos. In the proposed model, we decompose the host image into four normal sub-bands according; LL, LH, HL, HH by level one DWT, the high frequency (LH, HL and HH ) sub-bands are suitable for watermark embedding, as embedding watermark in LL sub-band causes the deterioration of image quality [47], the HL sub-band has been adopted for this model since it contains a mixture of both high and low frequency contents.

## 3.2 Discrete shearlet transform (DST)

Recently, many of the multi-scale transforms appeared and became widely applied in image processing, such as the curvelets, contourlets and Shearlets. These transforms merge between the multi-scale analysis and directional wavelets to find an optimal directional representation. Generally these are adopted by building a pyramid waveform which consists of different directions as well different scales. The nature of transforms architecture enables us to use directions in multi-scale systems. For this reason, these transforms are widely used in many applications such as sparse image processing applications, operators decomposition, inverse problems, edge detection, and image restoration [19].

In this study, we utilize the discrete shearlet transform which has many advantages, the most important is that it is a simple mathematical construct, where it depends on the affine systems theory, the best solution for sparse multidimensional representation [35]. Further, Shear transform applied to a fixed function, and exhibit the geometric and mathematical properties like directionality, elongated shapes, scales, oscillations [31]. With host image A, the Shearlet transform for the transformed output image B are computed by Laplacian pyramid scheme and directional filtering according to equation below,

$$ShImg \to SH\{HostImg(a,s,t)\}, \qquad (1)$$

where, $a$ is the scale parameter; $a > 0$, $s$ is the shear parameter or sometimes it called the direction; $s \in \mathbb{R}$, $t$ is the translation parameter or sometimes it called the location; $t \in \mathbb{R}$; $a$, $s$, and $t$ called Shearlet basis functions [33].

Shearlet transform are calculated by dilating, shearing and translation [33]. It is defined in equations below:

$$\begin{aligned} &SH\{HostImg(a,s,t)\} \\ &\qquad = \int HostImg(y)\,\Psi(x-y)\,dy \\ &\qquad = HostImg \times \Psi(x), \end{aligned} \qquad (2)$$

where $\Psi$ is $a$ generating function is computed by equation,

$$\psi(x) = |detA(a,s)| - 0.5\,\Psi(A(a,s-1)(x-t)) \qquad (3)$$

Where $a$ and $s$ are geometrical transforms and dilation, and are $2 \times 2$ matrices calculated by:

$$\mathbf{a} = \begin{bmatrix} a & 0 \\ 0 & \sqrt{a} \end{bmatrix}, \quad \mathbf{s} = \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix}, \qquad (4)$$

$$A(a,s) = \begin{bmatrix} a & s\sqrt{a} \\ 0 & \sqrt{a} \end{bmatrix}. \qquad (5)$$

DST is an appropriate way for image watermarking, that it achieved high performance for determine the directional features also the optimal localization [1]. In this paper, we used Finite Discrete Shearlet Transform[1] (FDST) for spread spectrum watermarking [24]. One of the main advantages of the FDST is that, it is only based on the Fast Fourier transform (FFT) for which very fast implementations are available. Further, using band-limited shearlets one can construct a Parseval frame that provides a simple and straightforward inverse shearlet transform. In the notations of the MATLAB software utilized here, the shearlet transform is applied to the image according to the following command:

```
[ST, Psi] = shearletTransformSpect(A,
numOfScales, realCoefficients);
```

This command returns two most important functions are ST and Psi, ST is the Shearlet coefficients as a three-dimensional matrix of size $M \times N \times \eta$ and Psi is the same size and contains the respective shearlet spectra. Each one of these functions stores 61 matrices and each matrix has the same size of the original image. According FDST toolbox described above, the matrix Psi(18) is considered good matrix for applying shearlet transform, where it is defined far from the external boundaries, which reduces the overlap.

## 3.3 Arnold transform (AT)

Arnold transform can be used to improve the security of the logo image that used in many watermarking applications [16, 55]. Suppose that $M$ is $n$-dimensional matrix $(n \times n)$ which represents the original image. Arnold transform can be calculated from $M$ as the following formula by equation,

$$\begin{bmatrix} x^* \\ y^* \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} (mod\ n), \qquad (6)$$

where $n$ is the size of the original image, $(x, y)$ are the original pixel coordinates, $(x^*, y^*)$ represents the coordinates of the pixel after applying Arnold transform. The principle of Arnold transform is based on modifying the location of the original pixels many times. This means the possibility of implementing it on the image periodically [46] which leads to the improvement of security in the watermarking image. Moreover, the mathematical characteristics [16] of the Arnold transform enables it to be used widely in the area of image watermarking. Note that the pixel location changes frequently, until it comes back to its original position after a number of iterations of Arnold transform, so the original image is recovered. The anti-Arnold transform which brings back to the original locations suffers from high time complexity that is needed in the reverse calculations [55].

[1]FDST is available as a MATLAB programs presented for applying DST. This software is available for free: http://www.mathematik.uni-kl.de/imagepro/software/ffst/

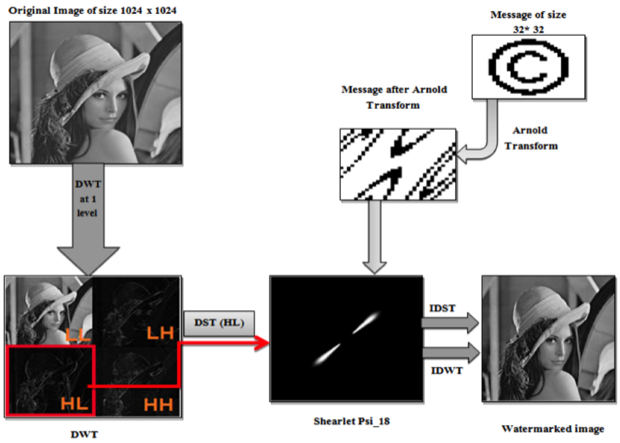


Figure 3: Watermark embedding algorithm of the proposed SWA method.

## 3.4 The hybrid SWA watermarking

A general digital image watermarking algorithm includes two procedures: (i) the first is the *embedding*, through this procedure the copyright image (message) hidden inside the original image (the host), the resulting image is called watermarked image [23], (ii) the second is the *extraction*, through this procedure the copyright image is extracted from the watermarked image. The proposed SWA watermarking model uses DWT, DST, and Arnold Transform for embedding procedure, in addition to using these for extraction procedure as well. Thus, our watermarking method consists of two major phases, (1) message embedding, and (2) message extraction, which we describe below in detail.

### 3.4.1 SWA embedding algorithm

Embedding algorithm procedure in the proposed SWA method consists of several steps, as shown in Figure 3. In the beginning, the host image (the original image) is being read, where its size is $(1024 \times 1024)$, also the copyright image is being read, where its size is $(32 \times 32)$.

The DWT applied to host image, where four of the sub-bands produced through applying this transform are; HH, HL, LH, and LL respectively. After that, the DST performed based on vertical frequency (HL) at level one DWT, From applying the DST produces two types of parameter $(\Psi$, ST) where each type consists of a 61 matrices, with the size of matrices to be $512 \times 512$.

The Arnold transform applied with copyright image, then each pixel of the resulting image after applying Arnold transform hide or embedded in the resulting host image obtained after applying DST. The embedding is performed at the matrix number (18) which resulted from applying DST at $\Psi$ parameter. The size of this matrix is $512 \times 512$, this matrix divided into 32 blocks (the size of matrix (18) by copyright image $(512/32)$, the pixel which obtained from applying Arnold embedded at the last pixel of each block

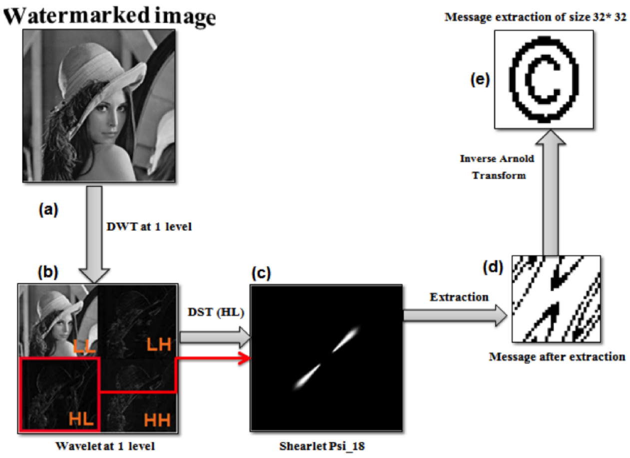


Figure 4: Watermark extracting algorithm of the proposed SWA method.

of the matrix (18), pixels $16 \times 16$, $16 \times 32$, etc. The following formula is calculated for all the values in the last Pixel from the matrix 18,

$$New\_value = SD(LPixl)^2/\alpha, \quad (7)$$

where (SD) is the standard deviation, $LPixl$ is last pixel in matrix 18 and value of $\alpha$ is set to 10000. From copyright image after applying Arnold, when the pixel of the ($New\_value$) is 1 then $New\_value$ is stored as it is, otherwise the negative of ($New\_value$) stored. Finally, inverse discrete wavelet transform (IDWT) and inverse discrete shearlet transform (IDST) are performed to return the image to normal shape.

### 3.4.2 Extraction algorithm SWA model

The steps to extract the message computed by the inverse procedure can be described as follows. The proposed extraction algorithm for the SWA watermarking model is depicted in Figure 4. Initially, the watermarked image is being read, where its size is $1024 \times 1024$. DWT applied with watermarked image, After that, the DST performed based on horizontal frequency (HL) at level one DWT, the Extraction performed at the matrix number (18) which resulting from $\Psi$ parameter. The size of this matrix is $512 \times 512$, this matrix divided into 32 blocks, when the value of the last pixel of each block of the matrix (18) is positive, then return 1 otherwise return 0, Thus, the previous Arnold transformed image was generated, and finally inverse Arnold transform was performed to get the copyright image.

The pseudo codes given in Algorithm 1 and Algorithm 2 summarizes the proposed embedding and extraction algorithms for the proposed SWA watermarking approach.

**Algorithm 1** Embedding of SWA model

Input: Host image ($1024 \times 1024$) and copyright image ($32 \times 32$)
Output: Watermarked image of size ($1024 \times 1024$)

1: Apply 2D_DWT (Host image)
2: Output: (LL, LH, HL, HH)
3: Apply DST (HL at step 2)
4: Output: 61 matrices ($\Psi$) of size $512 \times 512$
5: Get matrix $\Psi_{18}$ (number 18 from matrix $\Psi$), block size $16 \times 16$
6: Apply Arnold Transform (Copyright image)
7: Let $k0$= - $(std(Host)^2)/\alpha$
8: Let $k1$= $(std(Host)^2)/\alpha$
9: Let $\Psi_{18}\_W(x,y)$= $\Psi_{18}(x,y)$
10: For each pixel from copyright image after Arnold transform (message vector (pixel)).
11: If (message vector (pixel) is 0) then $\Psi_{18}\_W(y + blocksize - 1, x + blocksize - 1)$=$k0$
12: Else $\Psi_{18}\_W(y + blocksize - 1, x + blocksize - 1)$ =$k1$
13: If (message vector is last pixel) Go To step 15
14: Next pixel (message vector (pixel++)) Go To step 11
15: Let $\Psi_{18}$(x, y) = $\Psi_{18}\_W(x,y)$
16: Let C= IDST ($\Psi$, ST)
17: Let Watermarked image= 2D_IDWT (LL, LH, C, HH)
18: Output Watermarked image of size ($1024 \times 1024$)
19: Calculate Performance evaluation measures for (Watermarked image)

# 4 Experimental results

## 4.1 Setup and error metrics

Our experimental results were conducted on a Toshiba laptop with Intel (R) Core i5-2450M processor with CPU 2.50 GHz, RAM 6.00 GB in MATLAB environment. Most of the studies in the digital image watermarking literature are based on standard images such as Lena, Baboon, Boat, Cameraman, Peppers and Barbara images etc. These images taken from USC-SIPI miscellaneous database[2], it consist of 44 images, 16 colors and 28 monochromes. The sizes of images are $256 \times 256$, $512 \times 512$, and $1024 \times 1024$.

We gauge the performance of different watermarking techniques quantitatively by using six most common error metrics utilized in image processing literature which are given below.

– **Mean Square Error (MSE)**: MSE [27, 52] between original image and watermarked image is calculated using the formula:

$$MSE = \frac{1}{N} \sum_i \sum_j (f(i,j) - g(i,j))^2,$$

where sum $j$ and $k$ is taken over all the pixels in the image, $N$ is the total number of pixels in the image.

[2]http://sipi.usc.edu/database/database.php?volume=misc

**Algorithm 2** Extraction of SWA model

Input: Watermarked image ($1024 \times 1024$)
Output: Message image ($32 \times 32$)

1: Apply 2D_DWT (Watermarked image)
2: Output: (LL_W, LH_W, HL_W and HH_W).
3: Apply DST (HL_W at step 2).
4: Output: 61 matrices ($\Psi\_W$), 61 matrices (ST_W) size of each matrix is $512 \times 512$
5: Get matrix $\Psi_{18}\_W$ (number of 18 from matrix $\Psi\_W$), block size $16 \times 16$
6: For each last element from block at $\Psi_{18}\_W$ matrix
7: If $(\Psi_{18}\_W(y + blocksize - 1, x + blocksize - 1){=}negativevalue)$ then (message vector(pixel)=0)
8: Else (message vector(pixel)=1)
9: If ($\Psi_{18}\_W$ is last block) Go To step 10.
10: Message_EX= reshape(message vector(pixels))
11: Go To step 6
12: Apply Inverse Arnold Transform (Message_EX) .
13: Output Message_EX image.
14: Calculate Performance evaluation measures for (Message_EX image).
15: End

It is worth mentioning here that a good watermarking system should satisfy minimum MSE.

– **Root Mean Square Error (RMSE)**: RMSE [9] equals the square root of Mean Square Error ($MSE^{0.5}$); it can also be calculated using the formula,

$$RMSE = \sqrt{\frac{1}{N}\sum_i \sum_j (f(i,j) - g(i,j))^2}.$$

It is worth mentioning here that a good watermarking system should satisfy minimum RMSE.

– **Peak Signal to Noise Ratio (PSNR)**: PSNR [56] is used for measuring the quality of the watermarked image and defined using the formula:

$$PSNR = 10\log_{10} \frac{(max)^2}{\frac{1}{m \times n}\sum_j \sum_k (f(i,j) - g(i,j))^2},$$

where, $m \times n$ is the image size, $(max)$ is the maximum value of the pixels values in the image $f$ is the host image, $g$ is the watermarked images. It is worthily mentioned here that a higher value of PSNR (dB) is good because it means that the ratio of signal to noise is higher.

– **Signal to Noise Ratio (SNR)**: SNR [59] is mainly used to measure the sensitivity of the image. It measures the power of a signal strength relative to the background noise. It is calculated by the formula:

$$SNR = \frac{P_{signal}}{P_{noise}}$$

Higher values of SNR (dB) shows better performance.

– **Structural Similarity (SSIM)**: Another popular measure for similarity comparison between two images is SSIM [54] with values in the range of $[0, 1]$, where $1$ is acquired when two images are identical. Mean structural similarity index is in the range $[0, 1]$ and is known to be a better error metric than traditional signal to noise ratio [54]. It is the mean value of the structural similarity (SSIM) metric[3]. The SSIM is calculated between two windows $\omega_1$ and $\omega_2$ of common size $N \times N$, and is given by,

$$\mathrm{SSIM}(\omega_1, \omega_2) = \frac{(2\mu_{\omega_1}\mu_{\omega_2} + c_1)(2\sigma_{\omega_1\omega_2} + c_2)}{(\mu^2_{\omega_1} + \mu^2_{\omega_2} + c_1)(\sigma^2_{\omega_1} + \sigma^2_{\omega_2} + c_2)}$$

where $\mu_{\omega_i}$ the average of $\omega_i$, $\sigma^2_{\omega_i}$ the variance of $\omega_i$, $\sigma_{\omega_1\omega_2}$ the covariance, and $c_1, c_2$ stabilization parameters. The MSSIM value near $1$ implies the optimal denoising capability of a method and we used the default parameters.

– **Mean Absolute Error (MAE)**: This measure is used to compute the average magnitude of the errors in a set of forecasts, without considering their direction. It uses to compute the accuracy with continuous values [25]. MAE is calculated using the formula:

$$MAE = \frac{1}{N}\sum_i \sum_j |f(i,j) - g(i,j)|$$

Lower values of MAE indicates better performance.

## 4.2 Attacks

With the development of our proposed SWA watermarking technology, it is necessary to protect the images against several different types of attacks, which they are exposed to, such as compression, Gaussian filter, pepper and salt noise, median filter, cropping, resize, and rotation attacks. A watermarking method is said to be robust against some attack if we can recover the original image after that particular attack. Some of the common types of attack will be briefly explained below.

– **JPEG compression attack**: JPEG compression [60] and [3], is aimed to reduce the size of an image, this resize operation enables the users to upload and download images efficiently. Moreover, compression reduces the complexity time of sending multimedia material.

– **Salt and pepper noise attack**: Another type of attacks is Salt and pepper noise [12] and [30], which affected watermarking images. This kind of noise is not

[3]We use the default parameters for SSIM and the MATLAB code is available online at https://ece.uwaterloo.ca/ z70wang/research/ssim/.

only damaging the image through software used for that purpose, but it can also happen during the process of image acquisition, by affecting the used camera, or in the stage of storing in memory. Regarding to 8 bit grayscale image, salt and pepper noise randomly alter the pixel value to either minimum (0) or maximum ($2^8 - 1 = 255$).

– **Cropping attack**: Image cropping is the operation of cutting the outer part of the original image with specific measurements, to improve the image or get rid of the excess parts. This operation can be carried out using different block size ratio [60].

– **Median filter attack**: Mean filter is an image processing technique which changes the center value of a block (for example $3 \times 3$ block) of an image with the median value of the pixels, this leads to smooth image and reduces the variation of the density between any pixel and its neighbors. The main challenge of median filter attack [32], is the use of small number of pixels to reconfigure the original image

– **Rotation attack**: The principle of image rotation is the inverse transformation for each pixel of the original image, so the image is calculated using the interpolation [43]. Rotation attack could affect the image to various degrees based on the image rotation angle [21].

– **Resize (scaling) attack**: Image scaling Usually used to resize digital images for printing purposes, which does not change the actual pixels in the original image [14], But to keep the image without affected by scaling, users must take into account not to reduce the original image to less than half, or not to enlarge it to more than double as proven in studies [36]. Generally there two types of scaling attack; down-scaling attack and up-scaling attack [36].

– **Gaussian filter noise**: Gaussian filter is a geometric method which modify the original image by removing high frequency pixels [3], to reduce the amount of pixel density variation with the adjacent pixels, this can be done according to a specific formula depends on variance and mean [51]. The noise which caused by Gaussian filter resulted from adding random noise values to the actual pixel.

## 4.3 Detailed results

Our main experiments are divided into three parts each consist of multiple sub-experiments. The first part is comparison of our proposed SWA watermarking method with four common transforms based watermarking approaches available in the literature.

1. **DWT** - [57, 15, 37, 48, 58], and [7],
2. **DCT** - [44, 34, 49], and [10].
3. **DWT_DCT_ Joint** [2, 28, 5], and [6],
4. **DWT_ Arnold** - [55, 16] and [29].

All these techniques were implemented in MATLAB. In these experiments, we verify the performance improvements when we applied the proposed algorithm.

The second part is a comparison of quantitative results of our proposed SWA model, and four different approaches with seven attacks on $Lena$ image based on PSNR (dB), MSE, RMSE, SNR (dB), SSIM, and MAE error metrics. We also provide the ranking of these methods with respect to each of these error metrics.

The third part is a new way to compare different watermarking methods performances by combining multiple attacks together. These multi-attacks are applied on $Lena$ image and we compute the PSNR (dB), and SSIM error metrics as representative benchmarking of four methods from the literature with our proposed SWA watermarking method.

### 4.3.1 Comparison with other methods on multiple standard test images

Table 1 shows a comparison between our proposed approach with four of state-of-the-art watermarking approaches on multiple USC-SIPI standard test images. In order to test the advantage of the proposed SWA approach against these methods from the literature, we utilized six of the standard test images widely used (as a host image), and the copyright image which was used as embedded image.

As can be seen by comparing the different error metrics reported in Table 1 (with no attacks), the proposed SWA model achieved the best results across different images. The MSE measure with Boat image, the percentage of squares errors between the original image and the image, after embedding is (0.0015), and this low percentage indicates that our proposed approach obtains good result for the embedding step. Nevertheless, DWT_DCT Joint method obtained a decent result, it achieved (6.2047) whereas DWT achieved (104.8672), which shows that this method is not satisfactory. Similarly, outcomes of PSNR, which refers to the ratio of the noise signal of the image, as when the values of this measure are high, the quality of the image after embedding is good, the proposed method got the highest result (76.4063) and the lowest value presented when applying DWT method, the result was (27.9584). The MAE, which expresses the absolute error value between the original image and the embedded image, whenever the value of this measure was low the quality watermarked image is better. The proposed method got least absolute error value (0.0797) for the Lena image, while DWT method achieved the highest absolute error (8.1849). The SNR, which indicates confusion of the signals between the original image and the watermarked image, the perfect result was obtained with the proposed method (0.000) across all test images. The SSIM, which refers to the amount of similarity between the structure of the original image and

| Image/Methods | MSE | RMSE | PSNR | MAE | SNR | SSIM |
|---|---|---|---|---|---|---|
| Lena | | | | | | |
| DWT | 105.0586/5 | 10.2498/5 | 27.9505/5 | 8.1849/5 | -0.0258/5 | 0.5839/5 |
| DCT | 25.4757/4 | 5.0473/4 | 34.1035/4 | 4.0007/4 | -0.0058/4 | 0.8388/4 |
| DWT_DCT_Joint | 5.9893/3 | 2.4473/3 | 40.3910/3 | 0.9294/2 | -0.0014/2 | 0.9532/3 |
| DWT_ Arnold | 4.6186/2 | 2.0903/2 | 41.7607/2 | 1.2615/3 | 0.0018/3 | 0.9721/2 |
| Our SWA model | **0.0085/1** | **0.0919/1** | **68.8949/1** | **0.0797/1** | **0.0000/1** | **1.0000/1** |
| Baboon | | | | | | |
| DWT | 104.9922/5 | 10.2466/5 | 27.9532/5 | 8.1817/5 | -0.0247/4 | 0.8225/5 |
| DCT | 57.1129/4 | 7.5573/4 | 30.5975/4 | 5.0679/4 | -0.0060/3 | 0.9160/4 |
| DWT_DCT_Joint | 11.5060/2 | 3.3920/2 | 37.5556/2 | 1.1783/2 | -0.0017/2 | 0.9915/2 |
| DWT_ Arnold | 34.6575/3 | 5.8871/3 | 32.7668/3 | 4.1454/3 | 0.0348/5 | 0.9650/3 |
| Our SWA model | **0.0199/1** | **0.1410/1** | **65.1835/1** | **0.1152/1** | **0.0000/1** | **1.0000/1** |
| Barbara | | | | | | |
| DWT | 103.9729/5 | 10.1967/5 | 27.9956/5 | 8.1256/5 | -0.0303/5 | 0.6931/5 |
| DCT | 28.8601/4 | 5.3722/4 | 33.5618/4 | 4.1810/4 | -0.0074/3 | 0.8816/4 |
| DWT_DCT_Joint | 10.0737/2 | 3.1739/2 | 38.1329/2 | 1.0764/2 | -0.0019/2 | 0.9694/2 |
| DWT_ Arnold | 20.8456/3 | 4.5657/3 | 34.9747/3 | 2.6838/3 | 0.0269/4 | 0.9633/3 |
| Our SWA model | **0.0079/1** | **0.0890/1** | **69.1725/1** | **0.0702/1** | **0.0000/1** | **1.0000/1** |
| Cameraman | | | | | | |
| DWT | 101.1671/5 | 10.0582/5 | 28.1144/5 | 8.0435/5 | -0.0245/5 | 0.5771/5 |
| DCT | 23.7020/4 | 4.8685/4 | 34.4170/4 | 3.8271/4 | -0.0051/3 | 0.8375/4 |
| DWT_DCT_ Joint | 5.1074/2 | 2.2599/2 | 41.0828/ | 0.8677/2 | -0.0012/2 | 0.9482/2 |
| DWT_ Arnold | 20.5353//3 | 4.5316/3 | 35.0398/3 | 2.2142/3 | 0.0096/4 | 0.9299/3 |
| Our SWA model | **0.0161/1** | **0.1268/1** | **66.0995/1** | **0.1015/1** | **0.0000/1** | **0.9999/1** |
| Peppers | | | | | | |
| DWT | 102.5565/5 | 10.1270/5 | 28.0552/5 | 8.0597/5 | -0.0319/5 | 0.6091/5 |
| DCT | 27.3814/4 | 5.2327/4 | 33.7902/4 | 4.1240/4 | -0.0075/4 | 0.8457/4 |
| DWT_DCT_Joint | 9.7729/2 | 3.1262/2 | 38.2646/2 | 1.0404/2 | -0.0019/2 | 0.9603/2 |
| DWT_ Arnold | 11.4553/3 | 3.3846/3 | 37.5747/3 | 2.0799/3 | 0.0036/3 | 0.9392/3 |
| Our SWA model | **0.0089/1** | **0.0943/1** | **68.6709/1** | **0.0823/1** | **0.0000/1** | **1.0000/1** |
| Boat | | | | | | |
| DWT | 104.8672/5 | 10.2405/5 | 27.9584/5 | 8.1716/5 | -0.0214/5 | 0.6388/5 |
| DCT | 27.5275/4 | 5.2467/4 | 33.7671/4 | 4.1335/4 | -0.0051/4 | 0.8586/4 |
| DWT_DCT_Joint | 6.2047/2 | 2.4909/2 | 40.2376/2 | 0.9238/2 | -0.0011/2 | 0.9478/3 |
| DWT_ Arnold | 8.7440/3 | 2.9570/3 | 38.7477/3 | 1.7856/3 | 0.0042/3 | 0.9615/2 |
| Our SWA model | **0.0015/1** | **0.0387/1** | **76.4063/1** | **0.0323/1** | **0.0000/1** | **1.0000/1** |

Table 1: Comparison results of our proposed SWA model and four different approaches without attack on embedded copyright image. We show different error metric values for each method along with ranks. Best results are given in **boldface**.

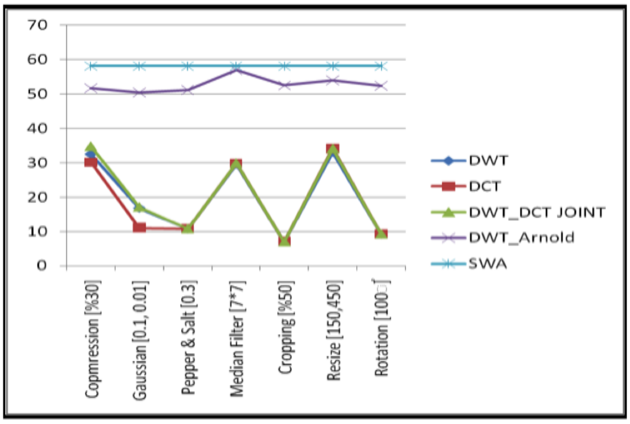


Figure 5: Graph showing the comparison of our proposed SWA model with other four methods based on PSNR (dB) values for the value of each type of attack on Lena image and copyright image.

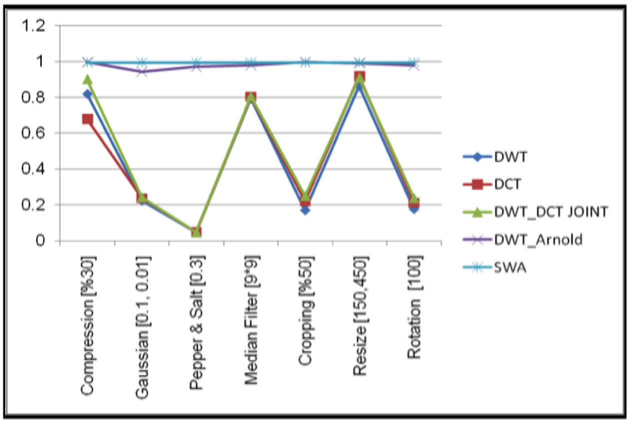


Figure 6: Graph showing the comparison of our proposed SWA model with other four methods based on SSIM values for the value of each type of attack on Lena image and copyright image.

the image after embedding, the proposed method got the highest results (1.000) or close to optimal.

### 4.3.2 Comparison of different attacks on Lena image with various error metrics

We next compare our proposed SWA model with other approaches with different attack methods under various error metrics. The watermarked image is exposed to several types of attacks with different parameter values as indicated appropriately. The compression attack expresses the ratio maintaining the image quality, for instant when the compression ratio is 10% which means that 90% of the image quality may be lost, with the maintaining 10% of the quality of the image, while this may not be discernible to the naked human eyes. Similarly, when the compression ratio is 90% means the 90% of the image quality has been preserved. For Gaussian noise the parameters indicate the mean and standard deviations indicating the amount of noise added to the image. Pepper and salt is a multiplicative noise with the probabilities given. Mean filtering is applied using the window size. Cropping uses the percentage of crop applied to the image. Resize is given in terms of the final size values. Rotation is performed at the angles given.

Figure 5 and Figure 6 shows the comparison of different attacks with respect to the PSNR (dB), and SSIM error metrics respectively. From the figures it is clear that the proposed SWA model performs the best with DWT_Arnold performing the next best. The results with other DWT, DCT, DWT_DCT_Joint perform rather poorly.

Table 2 shows the PSNR (dB) values, which is used to measure the quality of the image after embedding, comparing all transform techniques with our proposed method. When (10%) compression ratio is applied, the proposed SWA model achieved a high PSNR = 58.0975 dB value, and the worst result is achieved by DCT, with PSNR = 30.7130 dB. Except under median filtering our proposed SWA outperforms the other transform based approaches with many different types of attacks with various parameter settings.

Table 3 examines the impact of applying different methods on seven types of attack using Lena image with respect to the mean square error (MSE) measure. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (0.1016) error value. Although, it is clear from the results that DWT_ Arnold algorithm was better than our when we apply compression attack, except 10% compression ratio. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation, while the results proved the efficiency of DWT_ Arnold method with median and cropping attacks, but, with a little difference.

Similar observations can be made about RMSE metric on different attacks. Table 4 examines the impact of applying different methods on seven types of attack using Lena image with respect to the root mean square error (RMSE) measure. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (0.3187) error value. Although, it is clear from the results that DWT_Arnold algorithm was better than our when we apply compression attack, except (10%) compression ratio. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation, while the results proved the efficiency of DWT_Arnold method with median and cropping attacks, but, with a little difference.

Next, Table 5 investigates the impact of applying different methods on seven types of attack using Lena image with respect to signal to noise ratio (SNR) error metric. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (-0.0629) error value. Although, the DWT_Arnold algorithm was better than ours when we apply compression attacks in (80%), and (90%) compression ratio, and

| Attack/Methods | | DWT | DCT | DWT_DCT_Joint | DWT_Arnold | Our SWA |
|---|---|---|---|---|---|---|
| Compression | [10] | 30.7969/5 | 30.7130/4 | 30.9133/3 | 51.7164/2 | **58.0975/1** |
| | [30] | 32.4773/4 | 30.1909/5 | 34.7766/3 | 51.6497/2 | **58.0975/1** |
| | [50] | 30.4796/5 | 32.7898/4 | 35.5173/3 | 51.6591/2 | **58.0975/1** |
| | [70] | 28.0910/5 | 32.7397/4 | 35.3425/3 | 51.6591/2 | **58.0975/1** |
| | [90] | 27.4059/5 | 33.4065/4 | 38.0909/3 | 51.6591/2 | **58.0975/1** |
| Gaussian Noise | [0.03, 0.003] | 22.6204/5 | 23.7255/4 | 24.0203/3 | 51.0994/2 | **58.0975/1** |
| | [0.09, 0.009] | 17.4078/5 | 17.6835/4 | 17.7568/3 | 50.6410/2 | **58.0975/1** |
| | [0.1, 0.01] | 16.7995/4 | 11.0168/5 | 17.1067/3 | 50.3720/2 | **58.0975/1** |
| | [0.3, 0.03] | 8.1811/5 | 8.1858/4 | 10.1418/3 | 50.5888/2 | **58.0975/1** |
| | [0.5, 0.05] | 7.2834/5 | 7.4875/4 | 7.4862/3 | 50.7320/2 | **58.0975/1** |
| Pepper and Salt | [0.01] | 23.5386/5 | 24.9051/4 | 25.3305/3 | 51.7357/2 | **58.0975/1** |
| | [0.05] | 18.0867/5 | 18.4106/4 | 18.5091/3 | 51.5284/2 | **58.0975/1** |
| | [0.09] | 15.7175/5 | 15.9571/4 | 15.9726/3 | 51.2608/2 | **58.0975/1** |
| | [0.3] | 10.7093/5 | 10.7678/4 | 10.7559/3 | 51.1582/2 | **58.0975/1** |
| | [0.5] | 8.5105/5 | 8.5513/3 | 8.5415/4 | 51.1078/2 | **58.0975/1** |
| Median Filter | [1×1] | 27.9228/5 | 34.1035/4 | 40.3910/3 | **58.7254/1** | 58.0975/2 |
| | [3×3] | 33.3072/5 | 34.9873/4 | 36.1247/3 | **58.6774/1** | 58.0975/2 |
| | [5×5] | 31.3862/5 | 31.9093/4 | 32.1208/3 | **58.4451/1** | 58.0975/2 |
| | [7×7] | 29.5287/5 | 29.8056/4 | 29.8890/3 | 56.9006/2 | **58.0975/1** |
| | [9×9] | 28.2176/5 | 28.4383/4 | 28.4967/3 | 51.8829/2 | **58.0975/1** |
| Cropping | [10] | 5.7366/4 | 5.7368/3 | 5.7368/3 | 51.8929/2 | **58.0975/1** |
| | [30] | 6.11805/5 | 6.1198/4 | 6.1205/3 | 52.4585/2 | **58.0975/1** |
| | [50] | 6.9294/5 | 6.9355/4 | 6.9379/3 | 52.4245/2 | **58.0975/1** |
| | [70] | 8.4311/5 | 8.4487/4 | 8.4543/3 | 52.1612/2 | **58.0975/1** |
| | [90] | 13.0356/5 | 13.1217/4 | 13.1446/3 | 52.2802/2 | **58.0975/1** |
| Resize | [100,300] | 31.0705/5 | 31.2293/2 | 31.1997/3 | 52.3018/2 | **58.0975/1** |
| | [150,450] | 33.0241/5 | 34.1921/3 | 34.1008/4 | 53.8430/2 | **58.0975/1** |
| | [200,600] | 33.6675/5 | 36.1633/4 | 36.2744/3 | 55.7881//2 | **58.0975/1** |
| | [250,750] | 33.9085/5 | 36.9390/4 | 37.9065/3 | 57.1619/2 | **58.0975/1** |
| | [300,900] | 33.7695/5 | 36.8726/4 | 39.1607/3 | 57.2298/2 | **58.0975/1** |
| Rotation | [25] | 8.2973/5 | 8.3022/4 | 8.3050/3 | 52.9275/2 | **58.0975//1** |
| | [70] | 8.5009/5 | 8.5064/4 | 8.5105/3 | 51.7745/2 | **58.0975/1** |
| | [100] | 9.1657/5 | 9.1750/4 | 9.1810/3 | 52.3239/2 | **58.0975/1** |
| | [200] | 8.2427/5 | 8.2487/4 | 8.2517/3 | 52.8147/2 | **58.0975/1** |
| | [300] | 8.0150/5 | 8.0195/4 | 8.0217/3 | 52.4019/2 | **58.0975/1** |

Table 2: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on PSNR (dB) metric with ranks. Best results are given in **boldface**.

| Attack/Methods | | DWT | DCT | DWT_DCT_Joint | DWT_Arnold | Our SWA |
|---|---|---|---|---|---|---|
| Compression | [10] | 5.2654/5 | 4.6638/3 | 5.2194/4 | 0.1367/2 | **0.1016/1** |
| | [30] | 4.5640/4 | 6.3203/5 | 3.2932/3 | **0.0928/1** | 0.1016/2 |
| | [50] | 5.8867/5 | 4.5083/4 | 3.0358/3 | **0.0918/1** | 0.1016/2 |
| | [70] | 7.9038/5 | 4.6638/4 | 3.1663/3 | **0.0879/1** | 0.1016/2 |
| | [90] | 8.7213/5 | 4.3617/4 | 2.2306/3 | **0.0879/1** | 0.1016/2 |
| Gaussian Noise | [0.03, 0.003] | 15.1372/5 | 13.3999/4 | 12.9221/3 | 0.4395/2 | **0.1016/1** |
| | [0.09, 0.009] | 28.2383/5 | 27.4299/4 | 27.2864/3 | 0.6592/2 | **0.1016/1** |
| | [0.1, 0.01] | 30.4388/5 | 29.7588/4 | 29.5494/3 | 0.6592/2 | **0.1016/1** |
| | [0.3, 0.03] | 70.9105/3 | 71.0183/5 | 71.0116/4 | 0.7363/2 | **0.1016/1** |
| | [0.5, 0.05] | 99.7833/3 | 99.8240/4 | 99.9451/5 | 0.7188/2 | **0.1016/1** |
| Pepper and Salt | [0.01] | 9.32670/5 | 5.21240/4 | 2.2142/3 | 0.1299/2 | **0.1016/1** |
| | [0.05] | 14.1145/5 | 10.1257/4 | 7.3025/3 | 0.3057/2 | **0.1016/1** |
| | [0.09] | 18.8807/5 | 14.9787/4 | 12.2862/3 | 0.3584/2 | **0.1016/1** |
| | [0.3] | 44.0520/5 | 41.2690/4 | 38.8275/3 | 0.4590/2 | **0.1016/1** |
| | [0.5] | 67.9147/5 | 65.6387/4 | 64.3678/3 | 0.4834/2 | **0.1016/1** |
| Median Filter | [1×1] | 88.1478/5 | 4.0007/4 | 0.9288/3 | **0.0879/1** | 0.1016/2 |
| | [3×3] | 3.96670/5 | 3.0514/4 | 2.2597/3 | **0.0889/1** | 0.1016/2 |
| | [5×5] | 4.18210/5 | 3.6768/4 | 3.3332/3 | **0.0938/1** | 0.1016/2 |
| | [7×7] | 4.86260/5 | 4.4538/4 | 4.2133/3 | 0.1338/2 | **0.1016/1** |
| | [9×9] | 5.56320/5 | 5.1595/4 | 4.9448/3 | 0.4248/2 | **0.1016/1** |
| Cropping | [10] | 123.8123/5 | 123.7751/4 | 123.7279/3 | 0.2656/2 | **0.1016/1** |
| | [30] | 114.3665/5 | 114.0056/4 | 113.6263/3 | **0.0977/1** | 0.1016/2 |
| | [50] | 96.12130/5 | 95.12180/4 | 94.06350/3 | **0.0684/1** | 0.1016/2 |
| | [70] | 69.48680/5 | 67.5075/4 | 65.7319/3 | **0.0820/1** | 0.1016/2 |
| | [90] | 29.8192/5 | 26.4739/4 | 23.8796/3 | **0.0498/1** | 0.1016/2 |
| Resize | [100,300] | 4.1504/5 | 3.9113/3 | 3.9644/4 | 0.3857/2 | **0.1016/1** |
| | [150,450] | 3.8638/5 | 2.9439/3 | 3.0264/4 | 0.2705/2 | **0.1016/1** |
| | [200,600] | 3.9695/5 | 2.5879/4 | 2.5079/3 | 0.1729/2 | **0.1016/1** |
| | [250,750] | 3.9839/5 | 2.5813/4 | 2.1687/3 | 0.1260/2 | **0.1016/1** |
| | [300,900] | 4.1283/5 | 2.7542/4 | 1.9236/3 | 0.1240/2 | **0.1016/1** |
| Rotation | [25] | 81.9561/5 | 81.8842/4 | 81.8376/3 | 0.3340/2 | **0.1016/1** |
| | [70] | 80.5659/5 | 80.4934/4 | 80.4498/3 | 0.4355/2 | **0.1016/1** |
| | [100] | 74.5938/5 | 74.5079/4 | 74.4594/3 | 0.3838/2 | **0.1016/1** |
| | [200] | 84.3401/5 | 84.2893/4 | 84.2609/3 | 0.3428/2 | **0.1016/1** |
| | [300] | 85.6902/5 | 85.6161/4 | 85.5812/3 | 0.3770/2 | **0.1016/1** |

Table 3: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on MSE metric with ranks. Best results are given in **boldface**.

| Attack/Methods | | DWT | DCT | DWT_DCT_Joint | DWT_Arnold | Our SWA |
|---|---|---|---|---|---|---|
| Compression | [10] | 7.3665/5 | 5.9055/3 | 7.2874/4 | 0.3698/2 | **0.3187/1** |
| | [30] | 3.1988/3 | 7.9195/5 | 4.6710/4 | **0.3046/1** | 0.3187/2 |
| | [50] | 7.7434/4 | 5.8715/5 | 4.2892/3 | **0.3030/1** | 0.3187/2 |
| | [70] | 10.0878/5 | 5.9055/4 | 4.3764/3 | **0.2965/1** | 0.3187/2 |
| | [90] | 10.9318/5 | 5.4691/4 | 3.1893/3 | **0.2965/1** | 0.3187/2 |
| Gaussian Noise | [0.03, 0.003] | 18.9023/5 | 16.7106/4 | 16.1062/3 | 0.6629/2 | **0.3187/1** |
| | [0.09, 0.009] | 34.4540/5 | 33.3613/4 | 33.1316/3 | 0.8119/2 | **0.3187/1** |
| | [0.1, 0.01] | 36.9761/5 | 35.9851/4 | 35.7148/3 | 0.8149/2 | **0.3187/1** |
| | [0.3, 0.03] | 79.7524/5 | 79.6753/4 | 79.6042/3 | 0.8581/2 | **0.3187/1** |
| | [0.5, 0.05] | 108.0238/3 | 108.044/4 | 108.0808/5 | 0.8478/2 | **0.3187/1** |
| Pepper and Salt | [0.01] | 16.8436/5 | 14.2325/4 | 13.8030/3 | 0.3604/2 | **0.3187/1** |
| | [0.05] | 31.8880/5 | 30.5307/4 | 30.4602/3 | 0.5529/2 | **0.3187/1** |
| | [0.09] | 41.6718/5 | 40.5566/4 | 40.6133/4 | 0.5978/2 | **0.3187/1** |
| | [0.3] | 74.6177/5 | 74.4625/4 | 74.0317/3 | 0.6775/2 | **0.3187/1** |
| | [0.5] | 95.9893/5 | 95.6146/4 | 95.8191/4 | 0.6953/2 | **0.3187/1** |
| Median Filter | [1×1] | 10.2124/5 | 5.0473/4 | 2.4467/3 | **0.2965/1** | 0.3187/2 |
| | [3×3] | 5.51580/5 | 4.5591/4 | 3.9998/3 | **0.2981/1** | 0.3187/2 |
| | [5×5] | 6.90620/5 | 6.4979/4 | 6.3416/3 | **0.3062/1** | 0.3187/2 |
| | [7×7] | 8.55710/5 | 8.2786/4 | 8.1997/3 | 0.3658/2 | **0.3187/1** |
| | [9×9] | 9.93440/5 | 9.6901/4 | 9.6252/3 | 0.6518/2 | **0.3187/1** |
| Cropping | [10] | 132.2546/5 | 132.2520/4 | 132.2506/3 | 0.5154/2 | **0.3187/1** |
| | [30] | 126.5733/5 | 126.4560/3 | 126.5357/4 | **0.3125/1** | 0.3187/2 |
| | [50] | 115.2859/5 | 115.2027/4 | 115.1709/3 | **0.2615/1** | 0.3187/2 |
| | [70] | 96.98050/5 | 96.78460/4 | 96.7225/3 | **0.2864/1** | 0.3187/2 |
| | [90] | 57.07360/5 | 56.51360/4 | 56.3652/3 | **0.2232/1** | 0.3187/2 |
| Resize | [100,300] | 7.1552/5 | 7.0271/3 | 7.0514/4 | 0.6211/2 | **0.3187/1** |
| | [150,450] | 5.7099/5 | 4.9961/3 | 5.0494/4 | 0.5201/2 | **0.3187/1** |
| | [200,600] | 5.3152/5 | 3.9818/4 | 3.9311/3 | 0.4158/2 | **0.3187/1** |
| | [250,750] | 5.1489/5 | 3.6416/4 | 3.2583/3 | 0.3549/2 | **0.3187/1** |
| | [300,900] | 5.2512/5 | 3.6695/4 | 2.8195/3 | 0.3522/2 | **0.3187/1** |
| Rotation | [25] | 98.4825/5 | 98.4311/4 | 98.3983/3 | 0.5779/2 | **0.3187/1** |
| | [70] | 96.2016/5 | 96.1435/4 | 96.0986/3 | 0.6600/2 | **0.3187/1** |
| | [100] | 89.1230/5 | 89.0206/4 | 88.9595/3 | 0.6195/2 | **0.3187/1** |
| | [200] | 99.0954/5 | 99.0390/4 | 99.0047/3 | 0.5855/2 | **0.3187/1** |
| | [300] | 101.7332/5 | 101.6871/4 | 101.6610/3 | 0.6140/2 | **0.3187/1** |

Table 4: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on RMSE metric with ranks. Best results are given in **boldface**.

| Attack/Methods | | DWT | DCT | DWT_DCT_Joint | DWT_Arnold | Our SWA |
|---|---|---|---|---|---|---|
| Compression | [10] | 0.0068/2 | 0.0071/4 | **0.0062/1** | -0.2669/5 | -0.0629/3 |
| | [30] | 0.0047/2 | 0.0096//3 | **0.00037/1** | -0.0224/4 | -0.0629/5 |
| | [50] | 0.0114/3 | 0.0035/2 | **-0.00011/1** | -0.0179/4 | -0.0629/4 |
| | [70] | 0.0236/4 | 0.0052/3 | 0.00180/2 | **0.0000/1** | -0.0629/5 |
| | [90] | 0.0929/5 | 0.0059/3 | 0.00150/2 | **0.0000/1** | -0.0629/4 |
| Gaussian Noise | [0.03, 0.003] | 0.5236/4 | 0.5069/3 | 0.5027/2 | -2.1947/5 | **-0.0629/1** |
| | [0.09, 0.009] | 1.4204/4 | 1.4076/3 | 1.4058/2 | -4.6177/5 | **-0.0629/1** |
| | [0.1, 0.01] | 1.5562/4 | 1.5474/3 | 1.5424/2 | -4.8161/5 | **-0.0629/1** |
| | [0.3, 0.03] | 3.6103/2 | 3.6132/3 | 3.6138/4 | -5.9191/5 | **-0.0629/1** |
| | [0.5, 0.05] | 4.6983/2 | 4.7012/3 | 4.7047/4 | -5.6487/5 | **-0.0629/1** |
| Pepper and Salt | [0.01] | 0.0607/3 | 0.0403/2 | **0.0383/1** | -0.1870/5 | -0.0629/4 |
| | [0.05] | 0.2006/4 | 0.1825/3 | 0.1803/2 | -1.3078/5 | **-0.0629/1** |
| | [0.09] | 0.3399/4 | 0.3255/3 | 0.3053/2 | -1.6464/5 | **-0.0629/1** |
| | [0.3] | 0.9922/4 | 0.9804/2 | 0.9850/3 | -2.5172/5 | **-0.0629/1** |
| | [0.5] | 1.4573/4 | 1.4037/2 | 1.5374/4 | -2.7211/5 | **-0.0629/1** |
| Median Filter | [1×1] | 0.0254/4 | 0.0058/3 | 0.0014/2 | **0.0000/1** | -0.0629/5 |
| | [3×3] | -0.0154/4 | -0.0147/3 | -0.0137/2 | **-0.0045/1** | -0.0629/5 |
| | [5×5] | -0.0377/4 | -0.0333/3 | -0.0317/2 | **-0.0269/1** | -0.0629/5 |
| | [7×7] | -0.0553/4 | -0.0485/3 | -0.0455/2 | -0.2527/5 | **-0.0629/1** |
| | [9×9] | -0.6910/4 | -0.0606/3 | -0.0561/2 | -2.0421/5 | **-0.0629/1** |
| Cropping | [10] | -19.0662/3 | -19.0817/5 | -19.07895/4 | -0.9658/2 | **-0.0629/1** |
| | [30] | -10.1601/3 | -10.1779/4 | -10.1840/5 | **-0.0179/1** | -0.0629/2 |
| | [50] | -5.9778/3 | -5.9973/4 | -6.00370/5 | 0.1232/2 | **-0.0629/1** |
| | [70] | -3.2358/3 | -3.2558/4 | -3.2615/5 | **0.0267/1** | -0.0629/2 |
| | [90] | -0.8318/3 | -0.8516/4 | -0.8563/5 | 0.2219/2 | **-0.0629/1** |
| Resize | [100,300] | **-0.0202/1** | -0.0204/2 | -0.0205/3 | -1.8058/5 | -0.0629/4 |
| | [150,450] | **-0.0096/1** | -0.0115/2 | -0.0116 | -1.0502/5 | -0.0629/4 |
| | [200,600] | **-0.0040/1** | -0.0071/2 | -0.0071/2 | -0.4660/4 | -0.0629/3 |
| | [250,750] | **-0.0013/1** | -0.0044/2 | -0.0052/3 | -0.2056/5 | -0.0629/4 |
| | [300,900] | **0.0008/1** | -0.0027/2 | -0.0040/3 | -0.2056/5 | -0.0629/4 |
| Rotation | [25] | -2.4793/3 | -2.4844/4 | -2.4871/5 | -1.4738/2 | **-0.0629/1** |
| | [70] | -2.1654/2 | -2.1707/3 | -2.1742/4 | -2.2096/5 | **-0.0629/1** |
| | [100] | -1.2720/2 | -1.2780/3 | -1.2817/4 | -1.7788/5 | **-0.0629/1** |
| | [200] | -2.1649/3 | -2.1708/4 | -2.1736//5 | -1.5179/2 | **-0.0629/1** |
| | [300] | -2.7311/3 | -2.7365/4 | -2.7391/5 | -1.7187/2 | **-0.0629/1** |

Table 5: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on SNR (dB) metric with ranks. Best results are given in **boldface**.

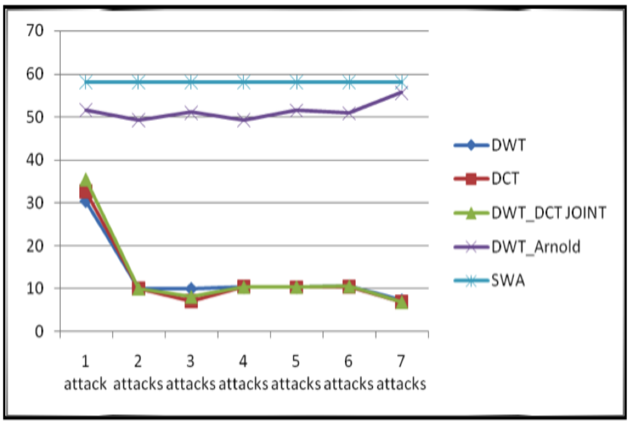


Figure 7: Graph showing the comparison of our proposed SWA model with other four methods based on PSNR (dB) values for multi-attacks on Lena image and copyright image.

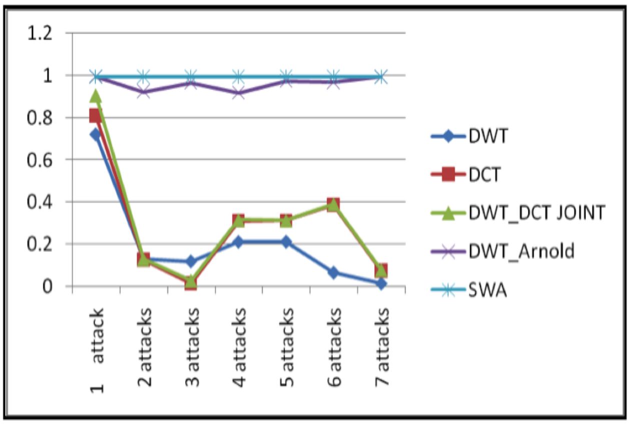


Figure 8: Graph showing the comparison of our proposed SWA model with other four methods based on SSIM values for multi-attacks on Lena image and copyright image.

DWT_DCT_Joint in (10%-50%) compression ratios. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation, while the results proved the efficiency of DWT_Arnold method with median and cropping attacks, but, with a little difference. Under Resize attack the DWT performed better under all sizes.

Perhaps the best error metric is SSIM which measures the performance in terms of preserving structural similarity between original and watermarking images. Table 6 investigates the impact of applying different methods on seven types of attack using Lena image with SSIM error metric. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (0.9950) similarity value. Although, it is clear from the results that DWT_Arnold algorithm was similar to our results when we apply compression attackss. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation, while the results proved the efficiency of DWT_Arnold method with median and cropping attacks, but, with a little difference.

Finally, Table 7 investigates the impact of applying different methods on seven types of attack using Lena image with respect to the mean absolute error (MAE) metric. The results show that our approach has achieved satisfactory results, and on average outperformed the rest of the methods with (0.1016) error value. Although, the DWT_Arnold algorithm was similar to our results when we apply compression attack, except (10%) compression ratio as well in Median Filter, and Cropping attacks. Results also proved the efficiency of our algorithm with various types of attacks such as Gaussian Noise, Salt and Peppers, Resizing and Rotation.

#### 4.3.3 Comparison of multi-attacks on Lena image with PSNR and SSIM error metrics

When an attack occurs in an image that may lead to the loss of information or quality of the image (extracted message) that will be extracted. The performance gauged by error metrics which are used to evaluate the efficiency of the algorithms used in the embedding and extraction processes are classified into two types: subjective techniques, which are based on a view of humans, and objective techniques. We selected the SSIM, and PSNR (dB) metrics as the subjective and objective representative error measures for the evaluation next. We expose the image to different number of the attacks sequentially. This is a new comparative method in the field of digital image watermarking with multi-attacks.

Table 8 presents these multi-attacks and their corresponding results for different watermarking methods. We performed different combinations of attacks and we started with the compression attack (with 50% ratio) applied on image first. As can be seen, the proposed SWA method achieved significantly superior results compared to the rest of methods with PSNR = 58.0975 dB, and SSIM = 0.9950. The DWT achieved the worst results among others with PSNR = 30.4512 dB, and SSIM = 0.7207. When the image was further exposed to noise and filtering attacks with random values, along with cropping, resize, rotation attacks, the proposed SWA method achieved the best results consistently with PSNR = 58.0975 dB, and SSIM = 0.9950. Note that the DCT, DWT_DCT_Joint methods obtained the worst results with PSNR = 6.9578, SSIM = 0.0773 and PSNR = 0.0792, respectively. These results indicate the robustness of our proposed SWA model against multi-attacks. Figure 7 and Figure 8 shows the comparison of sequential multi-attacks with respect to the PSNR (dB), and SSIM error metrics respectively. From the figures it is clear that the proposed SWA model performs the best with DWT_Arnold performing the next best. The results with other DWT, DCT, DWT_DCT_Joint perform rather poorly.

| Attack/Methods | | DWT | DCT | DWT_DCT_Joint | DWT_Arnold | Our SWA |
|---|---|---|---|---|---|---|
| Compression | [10] | 0.8271/5 | 0.8272/4 | 0.8297/3 | **0.9951/1** | 0.9950/2 |
| | [30] | 0.8204/5 | 0.6798/4 | 0.9031/3 | **0.9950/1** | **0.9950/1** |
| | [50] | 0.7204/5 | 0.8108/4 | 0.9062/3 | **0.9950/1** | **0.9950/1** |
| | [70] | 0.5985/5 | 0.7946/4 | 0.8820/3 | **0.9950/1** | **0.9950/1** |
| | [90] | 0.6004/5 | 0.8125/4 | 0.9259/3 | **0.9950/1** | **0.9950/1** |
| Gaussian Noise | [0.03, 0.003] | 0.3726/5 | 0.4320/4 | 0.4508/3 | 0.9751/2 | **0.9950/1** |
| | [0.09, 0.009] | 0.2376/5 | 0.2539/4 | 0.2599/3 | 0.9457/2 | **0.9950/1** |
| | [0.1, 0.01] | 0.2250/5 | 0.2390/4 | 0.2446/3 | 0.9428/2 | **0.9950/1** |
| | [0.3, 0.03] | 0.1322/5 | 0.1343/4 | 0.1368/3 | 0.9336/2 | **0.9950/1** |
| | [0.5, 0.05] | 0.1403/5 | 0.1427/4 | 0.1451/3 | 0.9338/2 | **0.9950/1** |
| Pepper and Salt | [0.01] | 0.4789/5 | 0.6486/4 | 0.7248/3 | 0.9949/2 | **0.9950/1** |
| | [0.05] | 0.2564/5 | 0.2937/4 | 0.3081/3 | 0.9877/2 | **0.9950/1** |
| | [0.09] | 0.1633/5 | 0.1750/4 | 0.1820/3 | 0.9826/2 | **0.9950/1** |
| | [0.3] | 0.0479/5 | 0.0487/4 | 0.0490/3 | 0.9704/2 | **0.9950/1** |
| | [0.5] | 0.0234/5 | 0.0238/4 | 0.0232/3 | 0.9704/2 | **0.9950/1** |
| Median Filter | [1×1] | 0.5839/5 | 0.8388/4 | 0.9532/3 | **0.9950/1** | **0.9950/1** |
| | [3×3] | 0.8477/5 | 0.9034/4 | 0.9264/3 | **0.9950//1** | **0.9950/1** |
| | [5×5] | 0.8525/5 | 0.8720/4 | 0.8810/3 | **0.9951/1** | 0.9950/2 |
| | [7×7] | 0.8214/5 | 0.8338/4 | 0.8392/3 | **0.9955/1** | 0.9950/2 |
| | [9×9] | 0.7950/5 | 0.8041/4 | 0.8087/3 | 0.9793/2 | **0.9950 /1** |
| Cropping | [10] | 0.0075/5 | 0.0085/4 | 0.0091/3 | 0.9901/2 | **0.9950/1** |
| | [30] | 0.0593/5 | 0.0765/4 | 0.0871/3 | **0.9958/1** | 0.9950/2 |
| | [50] | 0.1716/5 | 0.2206/4 | 0.2514/3 | **0.9965/1** | 0.9950/2 |
| | [70] | 0.3163/5 | 0.4249/4 | 0.4829/3 | **0.9958/1** | 0.9950/2 |
| | [90] | 0.4950/5 | 0.6946/4 | 0.7881/3 | **0.9962/1** | 0.9950/2 |
| Resize | [100,300] | 0.8609/5 | 0.8750/4 | 0.8722/3 | 0.9816/2 | **0.9950/1** |
| | [150,450] | 0.8621/5 | 0.9182/4 | 0.9122/3 | 0.9896/2 | **0.9950/1** |
| | [200,600] | 0.8450/5 | 0.9316/4 | 0.9318/3 | 0.9932/2 | **0.9950/1** |
| | [250,750] | 0.8368/5 | 0.9285/4 | 0.9444/3 | 0.9954/2 | **0.9950/1** |
| | [300,900] | 0.8248/5 | 0.9166/4 | 0.9528/3 | 0.9948/2 | **0.9950/1** |
| Rotation | [25] | 0.1498/5 | 0.1723/4 | 0.1875/3 | 0.9834/2 | **0.9950/1** |
| | [70] | 0.1534/5 | 0.1751/4 | 0.1952/3 | 0.9703/2 | **0.9950/1** |
| | [100] | 0.1784/5 | 0.2130/4 | 0.2398/3 | 0.9779/2 | **0.9950/1** |
| | [200] | 0.1459/5 | 0.1692/4 | 0.1851/3 | 0.9823/2 | **0.9950/1** |
| | [300] | 0.1337/5 | 0.1534/4 | 0.1651/3 | 0.9780/2 | **0.9950/1** |

Table 6: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on SSIM metric with ranks. Best results are given in **boldface**.

| Attack/Methods | | DWT | DCT | DWT_DCT_Joint | DWT_Arnold | Our SWA |
|---|---|---|---|---|---|---|
| Compression | [10] | 5.2654/5 | 4.6638/3 | 5.2194/4 | 0.1367/2 | **0.1016/1** |
| | [30] | 4.5640/4 | 6.3203/5 | 3.2932/3 | **0.0928/1** | 0.1016/2 |
| | [50] | 5.8867/5 | 4.5083/4 | 3.0358/3 | **0.0918/1** | 0.1016/2 |
| | [70] | 7.9038/5 | 4.6638/4 | 3.1663/3 | **0.0879/1** | 0.1016/2 |
| | [90] | 8.7213/5 | 4.3617/4 | 2.2306/3 | **0.0879/1** | 0.1016/2 |
| Gaussian Noise | [0.03, 0.003] | 15.1372/5 | 13.3999/4 | 12.9221/3 | 0.4395/2 | **0.1016/1** |
| | [0.09, 0.009] | 28.2383/5 | 27.4299/4 | 27.2864/3 | 0.6592/2 | **0.1016/1** |
| | [0.1, 0.01] | 30.4388/5 | 29.7588/4 | 29.5494/3 | 0.6592/2 | **0.1016/1** |
| | [0.3, 0.03] | 70.9105/4 | 71.0183/5 | 71.0116/3 | 0.7363/2 | **0.1016/1** |
| | [0.5, 0.05] | 99.7833/4 | 99.8240/5 | 99.9451/3 | 0.7188/2 | **0.1016/1** |
| Pepper and Salt | [0.01] | 9.32670/5 | 5.21240/4 | 2.2142/3 | 0.1299/2 | **0.1016/1** |
| | [0.05] | 14.1145/5 | 10.1257/4 | 7.3025/3 | 0.3057/2 | **0.1016/1** |
| | [0.09] | 18.8807/5 | 14.9787/4 | 12.2862/3 | 0.3584/2 | **0.1016/1** |
| | [0.3] | 44.0520/5 | 41.2690/4 | 38.8275/3 | 0.4590/2 | **0.1016/1** |
| | [0.5] | 67.9147/5 | 65.6387/4 | 64.3678/3 | 0.4834/2 | **0.1016/1** |
| Median Filter | [1×1] | 88.1478/5 | 4.0007/4 | 0.9288/3 | **0.0879/1** | 0.1016/2 |
| | [3×3] | 3.96670/5 | 3.0514/4 | 2.2597/3 | **0.0889/1** | 0.1016/2 |
| | [5×5] | 4.18210/5 | 3.6768/4 | 3.3332/3 | **0.0938/1** | 0.1016/2 |
| | [7×7] | 4.86260/5 | 4.4538/4 | 4.2133/3 | 0.1338/2 | **0.1016/1** |
| | [9×9] | 5.56320/5 | 5.1595/4 | 4.9448/3 | 0.4248/2 | **0.1016/1** |
| Cropping | [10] | 123.8123/5 | 123.7751/4 | 123.7279/3 | 0.2656/2 | **0.1016/1** |
| | [30] | 114.3665/5 | 114.0056/4 | 113.6263/3 | **0.0977/1** | 0.1016/2 |
| | [50] | 96.12130/5 | 95.12180/4 | 94.06350/3 | **0.0684/1** | 0.1016/2 |
| | [70] | 69.48680/5 | 67.5075/4 | 65.7319/3 | **0.0820/1** | 0.1016/2 |
| | [90] | 29.8192/5 | 26.4739/4 | 23.8796/3 | **0.0498/1** | 0.1016/2 |
| Resize | [100,300] | 4.1504/5 | 3.9113/4 | 3.9644/3 | 0.3857/2 | **0.1016/1** |
| | [150,450] | 3.8638/5 | 2.9439/4 | 3.0264/3 | 0.2705/2 | **0.1016/1** |
| | [200,600] | 3.9695/5 | 2.5879/4 | 2.5079/3 | 0.1729/2 | **0.1016/1** |
| | [250,750] | 3.9839/5 | 2.5813/4 | 2.1687/3 | 0.1260/2 | **0.1016/1** |
| | [300,900] | 4.1283/5 | 2.7542/4 | 1.9236/3 | 0.1240/2 | **0.1016/1** |
| Rotation | [25] | 81.9561/5 | 81.8842/4 | 81.8376/3 | 0.3340/2 | **0.1016/1** |
| | [70] | 80.5659/5 | 80.4934/4 | 80.4498/3 | 0.4355/2 | **0.1016/1** |
| | [100] | 74.5938/5 | 74.5079/4 | 74.4594/3 | 0.3838/2 | **0.1016/1** |
| | [200] | 84.3401/5 | 84.2893/4 | 84.2609/3 | 0.3428/2 | **0.1016/1** |
| | [300] | 85.6902/5 | 85.6161/4 | 85.5812/3 | 0.3770/2 | **0.1016/1** |

Table 7: Comparison results of our proposed SWA model and four different approaches with seven attacks on Lena image based on MAE metric with ranks. Best results are given in **boldface**.

| Attack | DWT | | DCT | | DWT_DCT_Joint | | DWT_Arnold | | Our SWA | |
|---|---|---|---|---|---|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
| Compr.50 | 30.4512 | 0.7207 | 32.7898 | 0.8108 | 35.5095 | 0.9063 | 51.6591 | 0.9950 | **58.0975** | **0.9950** |
| Gaussian Noise 0.3, 0.03 | 10.1372 | 0.1291 | 10.1321 | 0.1281 | 10.1422 | 0.1298 | 49.3246 | 0.9242 | **58.0975** | **0.9950** |
| Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 | 10.0453 | 0.1198 | 7.2584 | 0.0164 | 8.1910 | 0.0286 | 51.0662 | 0.9662 | **58.0975** | **0.9950** |
| Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 Median Filter $1 \times 1$ | 10.4129 | 0.2116 | 10.5051 | 0.3132 | 10.5011 | 0.3169 | 49.3081 | 0.9184 | **58.0975** | **0.9950** |
| Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 Median Filter $1 \times 1$ Cropping 30 | 10.4051 | 0.2123 | 10.4917 | 0.3140 | 10.5022 | 0.3140 | 51.5746 | 0.9743 | **58.0975** | **0.9950** |
| Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 Median Filter $1 \times 1$ Cropping 30 Resize 200,600 | 10.6769 | 0.0664 | 10.6127 | 0.3886 | 10.6494 | 0.3916 | 50.8800 | 0.9676 | **58.0975** | **0.9950** |
| Gaussian Noise 0.3, 0.03 Pepper & Salt 0.3 Median Filter $1 \times 1$ Cropping 30 Resize 200,600 Rotation 50 | 7.3280 | 0.0153 | 6.9578 | 0.0773 | 6.9597 | 0.0792 | 55.6671 | 0.9929 | **58.0975** | **0.9950** |

Table 8: Comparison results of our proposed SWA model and four different approaches with multi-attacks on Lena image based on PSNR (dB), SSIM error metrics. Compression (50%) attack was applied first and the remaining attacks are applied sequentially. Best results are given in **boldface**.

| Method | Embed. | Extr. | Total |
|---|---|---|---|
| DWT | 7.6934 | 5.1184 | 12.8118 |
| DCT | 1.9716 | 1.3098 | 3.2814 |
| DWT_DCT_Joint | 5.8207 | 3.1113 | 8.932 |
| DWT_Arnold | 1.1819 | 1.4375 | 2.6194 |
| Our SWA | 6.081 | 7.1644 | 13.2454 |

Table 9: Watermarking (embedding/extraction) time consumed (in seconds) by different approaches using Lena image and copyright message.

| Ref. | Approach | PSNR |
|---|---|---|
| [53] | Arnold | 63.25 |
| [45] | Arnold + DCT | 43.82 |
| [29] | DWT + Arnold | 62.79 |
| [22] | DWT + DCT | 57.67 |
| [35] | DWT+Shearlet | 67.54 |
| [18] | Entropy + Hadamard | 42.74 |
| [26] | Log-average luminance | 62.49 |
| [17] | DWT + DCT + SVD | 57.09 |
| [11] | DFT + 2D histogram | 49.45 |
| Our | DWT + DST + Arnold | **68.89** |

Table 10: Comparison of PSNR (dB) values of the watermarked Lena image using some recent methods with our proposed SWA method.

### 4.4 Timing and other watermarking methods comparison

In Table 9 we show the total time consumed (in seconds) for each methods compared here in terms of the embedding and extraction of the message from a $1024 \times 2014$ image. It can be noted that the proposed SWA consumed the highest amount of time particularly in the extraction phase, this is due to the use of three different types of transforms. The DW_Arnold transform based method takes overall the least amount of time.

Finally, in Table 10 we show the PSNR (dB) comparison results with some more recent studies available in the literature. Note that these methods also utilize the standard test Lena image, and the values indicate that our proposed SWA outperforms them with highest PSNR = 68.89 dB, followed by [35] which has PSNR = 67.54 dB, with pure Arnold transform based approaches fairing better than other transforms.

## 5 Conclusions and future works

Digital watermarking is an active area of research within security and there are many automatic systems that have been presented to secure the ownership information of the digital image based on watermarking. These systems utilized available techniques from image processing and data mining areas and apply them to digital images. In this paper, we studied a new hybrid model called SWA which is based on shearlet, wavelet, and Arnold transforms for efficient, and robust digital image watermarking. This model combines three transforms - discrete wavelet (DWT), discrete shearlet transform (DST), and Arnold transform - and their respective advantages. We utilized standard error image metrics to evaluate the proposed SWA method with seven types of attacks consists of different values of parameters; as well as multi-attacks which was used as a new way for comparing the effects of the attack on the image. Our results show that the proposed method is not affected by multi-attacks since applying shearlet at HL sub-band reduced the influence of watermarked image in any attack. Shearlet transform generates a series of matrices that obtained from the HL sub-band, where this sub band contains various features of the original image. In this way, the proposed SWA algorithm preserves a great deal of information. As a consequence, the attack is distributed to all of the shearlet derived matrices and the probability of changing the value of the pixel that has been embedding by the attack is very low. Based on this analysis, the results were stable or semi-static whatever the type of attack and its values.

Comparison results proved the robustness and strength of the proposed SWA method again other transform based state of the art methods. These results show that the proposed SWA model can be useful in securing image component in watermarking area. According to the results achieved by the proposed hybrid model, we consider that it is encouraging to apply this hybrid with several multimedia systems such as Video, text and audio, we expect that this system will achieve satisfactory results. Since the current research trends towards the multicore computing systems, our model may increase protection of videos transmission ownership.

Although the performance of the proposed method was the best comparing to state-of-the-art methods, it consumes more computational time than other approach, reducing this is one of the important future works. In this work, we applied shearlet transform at HL sub-band from level one in the wavelet transform, as another future work, we propose to go ahead towards applying shearlet transform with wavelet transform at the second and third level as well. Further, we can perform with other sub-bands such as LL, LH and HH to see if the robustness can be increased when certain types of attacks are applied. In our proposed method, we applied shearlet transform on the results of the wavelet transform, shearlet transform can also be applied with several other enhanced transforms like joint DWT with DCT etc.

## References

[1] B. Ahmederahgi, F. Kurugollu, P. Milligan, A. Bouridane (2013) Spread spectrum image watermarking based on the discrete shearlet transform, *4th Euro-*

*pean Workshop Visual Information Processing (EU-VIP)*, pp. 178–183.

[2] A. Al-Haj (2007) Combined DWT-DCT digital image watermarking, *Journal of computer science*, Vol. 3, pp. 740–746.

[3] Z. N. Y. Al-Qudsy (2011) An efficient digital image watermarking system based on contourlet transform and discrete wavelet transform *PhD Dissertation*, Middle East University, Turkey.

[4] Y. B. Amar, I. Trabelsi, N. Dey, M. S. Bouhlel (2016). Euclidean distance distortion based robust and blind mesh watermarking. International Journal of Interactive Multimedia and Artificial Inteligence, Vol. 4.

[5] S. K. Amirgholipour, A. R. Naghsh-Nilchi (2009) Robust digital image watermarking based on joint DWT-DCT, *International Journal of Digital Content Technology and its Applications*, pp. 42–54.

[6] R. Anju, Vandana (2013) Modified algorithm for digital image watermarking using combined DCT and DWT, *International Journal of Information and Computation Technology*, Vol. 3, No. 7, pp. 691–700.

[7] D. Baby, J. Thomas, G. Augustine, E. George, N. R. Michael (2015) A novel DWT based image securing method using steganography, *Procedia Computer Science*, No. 46, pp. 612–618.

[8] N. Y. Baithoon (2014) Zeros Removal with DCT Image Compression Technique, *Journal of Kufa for Mathematics and Computer* Vol. 1, No. 3.

[9] I. Belhadj, Z. Kbaier (2006) A novel content preserving watermarking scheme for multispectral images, *2nd International Conference on Information and Communication Technologies*, pp. 322–327.

[10] T. Bhaskar, D. Vasumathi (2013) DCT based watermark embedding into mid frequency of DCT coefficients using luminance component, *Elektronika ir Elektrotechnika*, Vol. 19, No. 4.

[11] F. G. M. Cedillo-Hernandez, M. Nakano-Miyatake, H. Manuel Pérez-Meana (2014) Robust hybrid color image watermarking method based on DFT domain and 2D histogram modification, *Signal, Image and Video Processing* Vol. 8, No. 1, pp. 49–63.

[12] C. H. V. Reddy, P. Siddaiah (2015) Medical image watermarking schemes against salt and pepper noise attack, *International Journal of Bio-Science and Bio-Technology* Vol. 7, No. 6, pp. 55–64.

[13] S. Chakraborty, S. Chatterjee, N. Dey, A. S. Ashour, A. E. Hassanien (2017). Comparative approach between singular value decomposition and randomized singular value decomposition-based watermarking. In Intelligent Techniques in Signal Processing for Multimedia Security (pp. 133-149). Springer.

[14] W. O. O. Chaw-Seng (2007) Digital image watermarking methods for copyright protection and authentication, *PhD Dissertation*, Queensland University of Technology, Australia.

[15] P.-Y. Chen, H.-J. Lin A (2006) DWT based approach for image steganography, *International Journal of Applied Science and Engineering*, Vol. 4, No. 3 pp. 275–290.

[16] M. F. M. El Bireki, M. F. L. Abdullah, M. Ali Abdrhman (2016) Digital image watermarking based on joint (DCT-DWT) and Arnold Transform, *International Journal of Security and Its Applications*, Vol. 10, No. 5, pp. 107–118.

[17] S. Fazli, M. Moeini (2016) A robust image watermarking method based on DWT, DCT, and SVD using a new technique for correction of main geometric attacks, *Optik-International Journal for Light and Electron Optics*, Vol. 2, No. 127, pp. 964–972.

[18] V. F. Rajkumar, G. R. S. Manekandan, V. Santhi (2011) Entropy based robust watermarking scheme using Hadamard transformation technique, *International Journal of Computer Applications* Vol. 12, No. 9.

[19] X. Gibert, V. M. Pate;, D. Labate, R. Chellappa (2014) Discrete shearlet transform on GPU with applications in anomaly detection and denoising, *EURASIP J. Adv. Signal Process*, pp. 1–14.

[20] B. L. Gunjal, R. R. Manthalkar (2010) An overview of transform domain robust digital image watermarking algorithms, *Journal of Emerging Trends in Computing and Information Sciences* Vol. 2, No. 1, pp. 37–42.

[21] X. Guo-juan, W. Rang-ding (2009) A blind video watermarking algorithm resisting to rotation attack, *International Conference on Computer and Communications Security (ICCCS).* Hong Kong, pp. 111–114.

[22] I. I. Hamid, E. M. Jamel (2016) Image watermarking using integer wavelet transform and discrete cosine transform, *Iraqi Journal of Science* Vol. 57, No. 2B, pp. 1308–1315.

[23] A. E. Hassanien, M. Tolba, and A. T. Azar (2014) Advanced machine learning technologies and applications. *Second International Conference*, Egypt, AML,Springer.

[24] S. Häuser, and G. Steidl (2012) Fast finite shearlet transform, *arXiv preprint*.

[25] K. K. Hiran, R. Doshi (2013) Robust & secure digital image watermarking technique using concatenation process, *International Journal of ICT and Management*

[26] J. A. Hussein (2010) Spatial domain watermarking scheme for colored images based on log-average luminance, *Journal of Computing* Vol. 2, no. 1.

[27] X. Kang, J. Huang, Y. Q Shi, Y. Lin (2003) A DWT-DFT composite watermarking scheme robust to both affine transform and JPEG compression, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 8, pp. 776–786.

[28] S. A. Kasmani, A. NaghshNilchi (2008) A new robust digital image watermarking technique based on joint DWT-DCT Transformation, *IEEE International Conference on Convergence and Hybrid Information Technology (ICCIT)*, pp. 539–544.

[29] R. Keshavarzian, A. Aghagolzadeh (2016) ROI based robust and secure image watermarking using DWT and Arnold map, *AEU-International Journal of Electronics and Communications*, pp. 278–288.

[30] S. K. A. Khalid, M. Mat Deris, and K. M. Mohamad (2013) A robust digital image watermarking against salt and pepper using sudoku, *The Second International Conference on Informatics Engineering and Information Science (ICIEIS)*.

[31] D. Labate, W.-Q. Lim, G. Kutyniok, G. Weiss (2005) Sparse multidimensional representation using shearlets. *Optics & Photonics*, pp. 59140U-59140U.

[32] J. C. Lee Analysis of attacks on common watermarking techniques.

[33] W.-Q. Lim (2010) The discrete shearlet transform: A new directional transform and compactly supported shearlet frames, *IEEE Transactions on Image Processing*, pp. 1166–1180.

[34] Lin, Shinfeng D., and C.-F. Chen (2000) A robust DCT-based watermarking for copyright protection, *IEEE Transactions on Consumer Electronics*, Vol. 3, No. 46, pp. 415–421.

[35] M. Mardanpour, Mohammad Ali Zare, Chahooki (2016) Robust hybrid image watermarking based on discrete wavelet and shearlet transforms, *arXiv preprint*.

[36] Y. Naderahmadian, S. Beheshti (2015) Robustness of wavelet domain watermarking against scaling attack, *IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE)*, Canada, pp. 1218–1222.

[37] N. Dey, R. A. Bardhan, D. Sayantan (2011) A novel approach of color image hiding using RGB color planes and DWT, *International Journal of Computer Applications*.

[38] N. Dey, M. Pal, A. Das (2012). A session based blind watermarking technique within the NROI of retinal fundus images for authentication using DWT, spread spectrum and Harris corner detection. arXiv preprint 1209.0053.

[39] N. Dey, B. Nandi, P. Das, A. Das, S. S. Chaudhuri (2013). Retention of electrocardiogram features insignificantly devalorized as an effect of watermarking for a multi-modal biometric authentication system. Advances in biometrics for secure human authentication and recognition, 175.

[40] N. Dey, S. Samanta, X. S. Yang, A. Das, S. S. Chaudhuri (2013). Optimisation of scaling factors in electrocardiogram signal watermarking using cuckoo search. International Journal of Bio-Inspired Computation, Vol. 5, No. 5, pp. 315–326.

[41] N. Dey, G. Dey, S. Chakraborty, S. S. Chaudhuri (2014). Feature analysis of blind watermarked electromyogram signal in wireless telemonitoring. In Concepts and Trends in Healthcare Information Systems (pp. 205-229). Springer.

[42] N. Dey, M. Dey, S. K. Mahata, A. Das, S. S. Chaudhuri (2015). Tamper detection of electrocardiographic signal using watermarked biohash code in wireless cardiology. International Journal of Signal and Imaging Systems Engineering, Vol. 8, No. 1-2, pp.46–58.

[43] F. O. Owalla, E. Mwangi (2012) A robust image watermarking scheme invariant to rotation, scaling and translation attacks, *16th IEEE Mediterranean Electrotechnical Conference*, Hammamet, pp. 379-382.

[44] A. Piva, B. Mauro, B. Franco, C. Vito (1997) DCT-based watermark recovering without resorting to the uncorrupted original image, *IEEE International Conference on Image Processing*, pp. 520–523.

[45] C. Pradhan, V. Saxena, A. K. Bisoi (2012) Imperceptible watermarking technique using Arnold's transform and cross chaos map in DCT Domain, *International Journal of Computer Applications*, Vol. 55, No. 15.

[46] N. Saikrishna, M. G. Resmipriya (2016) An invisible logo watermarking using Arnold transform, *Procedia Computer Science*, pp. 808–815.

[47] C.N. Sujatha, P. Satyanarayana (2015) Hybrid color image watermarking using multi frequency band, *International Journal of Scientific and Engineering Research*, Vol. 6, No. 1, pp. 948–951.

[48] S. Swami (2013) Digital image watermarking using 3 level discrete wavelet transform, *Conference on Advances in Communication and Control Systems*.

[49] T. Tewari, V. Saxena (2010) An improved and robust DCT based digital image watermarking scheme, *International Journal of Computer Applications*, Vol. 1, No. 3, pp. 28–32.

[50] Van Schyndel, Ron G., Andrew Z. Tirkel, and Charles F. Osborne (1994) A digital watermark, *IEEE International Conference on Image Processing*, Vol. 86–90.

[51] J. Veerappan, G. Pitchammal (2012) Interpolation based image watermarking resisting to geometrical attacks, *IEEE International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME)*, pp. 252–256.

[52] H.-J. Wang, P.-C. Su, and C-C. J. Kuo (1998) Wavelet-based digital image watermarking, *Optics Express*, Vol. 3, No. 12, pp. 491–496.

[53] H.-Q. Wang, J.-C. Hao, F.-M. Cui (2010) Colour image watermarking algorithm based on the Arnold transform, *IEEE International Conference on Communications and Mobile Computing (CMC)*, pp. 66–69.

[54] Z. Wang, E. P. Simoncelli, A. C. Bovik (2004) Multiscale structural similarity for image quality assessment, *Thirty-Seventh Asilomar Conference Signals, Systems and Computers*, pp. 1398–1402.

[55] L. Wu, J. Zhang, W. Deng, D. He (2009) Arnold transformation algorithm and anti-Arnold transformation algorithm, *IEEE International Conference on Information Science and Engineering*, pp. 1164–1167.

[56] Y. Wu, G. Xin, M. S. Kankanhalli, Z. Huang (2001) Robust invisible watermarking of volume data using the 3D DCT, *Computer Graphics International*, pp. 359–362.

[57] X. G. Xia, B. Charles, and G. Arce (1998) Wavelet transform based watermark for digital images, *Optics Express*, Vol. 3, No. 12, pp. 497–511.

[58] S. Zagade, S. Bhosale (2014) Secret data hiding in images by using DWT Techniques, *International Journal of Engineering and Advanced Technology* Vol. 3, no. 5.

[59] X. Y. Zhao, H. Wang (2006) A novel synchronization invariant audio watermarking scheme based on DWT and DCT, *IEEE Transactions on Signal Processing*, Vol. 54, No. 12, pp. 4835–4840.

[60] T.-R. Zong, Y. Xiang, S. Elbadry, S. Nahavandi (2016) Modified moment-based image watermarking method robust to cropping attack, *International Journal of Automation and Computing*, Vo. 13, No. 3, pp. 259–267.