International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-2, December, 2019

# A Dwt and Scrambling Based Blind Watermarking Algorithm for Digital Images Authentication

**Amira Eid, Ahmed A. Emran, Ahmed Y. Morsy**

***Abstract:*** ***Like the other multimedia that is spread on the Internet, images are also vulnerable to theft and attacks. Protecting the image is therefore an urgent necessity because it represents a large proportion of the digital content. Authentication and ownership protection are the basic demands of image security and these are achieved by applying watermarking techniques. For the Muslim world, the Holy Quran has its sanctity, which does not accept any controversy or doubt. As part of keeping pace with modern technology, digital copies of the Holy Qur'an are available, which are widely distributed all over the world. Therefore, it is necessary to ensure that these copies maintain their integrity and ensure that there are no malicious manipulations. In this paper, we propose an image watermarking scheme to authenticate the images of digital version of Holy Quran using discrete wavelet transform DWT. Here a fragile watermark is used to clarify whether there is any modification occurred to the intended images. Initially the cover image is decomposed by DWT where 2nd and 4th level coefficients are exploited for watermark embedding. The intended watermark is obtained by scrambling the original cover image. Then the scrambled image is inserted into the DWT coefficients by several trials using different embedding gains. To evaluate our system and see how effective it is to detect any error or manipulation, PSNR, SSIM and MSE are employed beside that they are acting as an imperceptibility measure. Results proved that our method has achieved a good level of imperceptibility and can detect any slight tamper. It is necessary to bear in mind that this method is valid for application to normal color images as well and gives an excellent level of efficiency.***

***Keywords: Holy Quran, Authentication, Discrete Wavelet Transform (DWT), Fragile watermark, Image watermarking, Scrambling.***

## I. INTRODUCTION

The worldwide expansion of the internet has contributed significantly to provide multimedia resources and digital data exchange in its various forms (image, text, audio and video). Despite the fact that the Internet has overcome many of the obstacles facing the web users by providing them with an inexhaustible source of digital information, but on the other hand that is accompanied by increased piracy, copyright violations and digital content manipulation [1]. Therefore, the property rights protection and data integrity ensure have imposed themselves as an imperative concerns. Images represent a very important aspect of human life all over the world. It can be used to express diverse forms of information. For the Muslim world, the ease to obtain a digital version of the Holy Quran is one of the most important advantages provided by the internet. This has facilitated the acquisition of a portable soft copy of the Quran in mobile phones in addition to other smart devices. However, this could expose the distributed Quran copies to malicious attempts by some saboteurs to mislead the believers and confuse them about some of the doctrinal things on which their faith is based. Therefore, ensuring the validity of soft copies of Quran is a non-negotiable subject that cannot be overlooked [2]. Reaching this goal is via applying the watermark technique. Given the very sensitive nature of the Qur'anic text, so it is better to treat Qur'anic pages as an image. Therefore, image watermarking is the most appropriate way to detect any manipulation while maintaining the quality of the watermarked text image at the same time. Authentication or tamper detection is realized by using a fragile watermarking algorithm. The more fragile the watermark becomes, the more likely it is to discover any little manipulation. Image watermarking algorithms can be categorized to the following denominations: spatial domain and transform domain [3]. In spatial domain watermarking, the watermark is inserted into the host image by directly changing predefined pixels values via bit permutation [4]. The number of bits which are modified inside the pixels should be considered otherwise, the watermark will be noticeable**.** Spatial watermarking algorithms are simple and easy to be implemented with low cost but, it doesn't guarantee a sufficient level of robustness as the watermark can be removed easily. For transform domain a robust watermarked image is obtained by exploiting the transform coefficients to hide the watermark. This technique withstands against attacks due to the fact that the secret information is spread in the whole spectrum [5]. Discrete Wavelet Transform (DWT) and discrete cousin transform (DCT) are considered as the most popular transforms. But DWT is dominant due to its better ability to localize time and frequency.

Revised Manuscript Received on December 30, 2019.
* Correspondence Author
**Amira Eid*,** Departement of Computer and Electrical Engineering, Higher Technological Institute HTI, 10th of Ramadan City, Egypt. Email: amira.eid@hti.edu.eg
**Ahmed A. Emran,** Department of Electrical Engineering, Al-Azhar University, Cairo, Egypt. Email: ahmedemran83@gmail.com
**Ahmed Y. Morsy,** Department of Electrical Engineering, Al-Azhar University, Cairo, Egypt. Email: ahmed_yahya_1@yahoo.com

The remainder of the paper is summarized as follows. Section II provides a summary of the methods and applications presented by previous researchers. Section III presents the proposed methodology while section IV provides system evaluation. Section V presents results and discussion while section VI concludes the paper and its future directions.

## II. LITERATURE REVIEW

Image authentication dilemma has received great attention in watermarking field. But most of algorithms focus on traditional images and few are interested in images of the Holy Scriptures texts. The Holy Quran occupies a very great position in the Islamic society. Modern technology has contributed to provide soft copies of the great book. Given the spiritual importance of the Qur'an, any malicious attempt to mislead the believers must be taken into account. Obviously, it is imperative to make sure that the widespread Quran copies are authentic and to detect any alteration. The previous methods are either very simple so it is easy to detect the presence of a watermark or so complex that it takes a great time to insert and extract the watermark. Since watermarking is applied to about 604 pages, it should be fast, effective and requires less information during the extracting process [6].

Kurniawan combines DWT with spatial domain for gray scale Quran images authentication [2]. Firstly DWT is applied to the host Quran image to get its wavelet components. The hash value of the original image is obtained and with the help of DWT coefficients. That matrix is used to alter the LSB of pixel values in the new image to produce a watermarked image. The same previous method was adopted in [7] in addition to applying the logarithm on images other than Quran images.

A non-blind robust scheme for gray scale image is implemented in [8]. This one select the most convenient blocks in the host image for watermark insertion. Then the DWT factors of each selected block is obtained where SVD coefficients of LL sub-band and the watermark signal are calculated to have the watermarked image.

Singh provides a contribution for identity authentication of medical images [9] by embedding both text and image watermarks. DWT, DCT and SVD are applied on cover image besides the watermark image whilst the text watermark is encrypted. The S component of watermark image is inserted in the S vector of the host image and the text mark is embedded in the diagonal component of second level of host image.

An approach to assure the integrity of Holy Quran verses is implemented in [10]. Here RIPEMD160 and SHA256 hash functions are employed integrity check in addition to using a compression technique for the Arabic letters. This technique requires the hash value for each single verse during authentication process besides each verse has two different hash values obtained from RIPEMD160 and SHA256. This makes the extraction process more complicated.

Another robust algorithm to protect gray scale images of Arabic text is employed by Alotaibi et al. [11]. This one combines both IWT and DCT to hide the gray scale watermark. First of all the host image is analyzed by IWT then DCT is applied to the lower side-band LL. Finally the watermark is embedded in the lower to medium coefficient of DCT.

In [12], spatial domain and transform domain are joined together and produced an algorithm for Holy Quran pages authentication and manipulation recovery. Discrete wavelet transform DWT is employed to decompose the cover image to its DWT coefficients and the watermark is embedded. Then the watermarked DWT coefficient is converted back to the spatial domain where another watermark is hidden via the least significant bits. A chaotic map is used to obscure the presence of a watermark and protect it from local attack. This algorithm provides a high level of imperceptibility and a high ability of tamper detection but this is at the expense of complexity plus it requires the existence of the host image during extraction process.

Olanrewaju presented a non-blind algorithm to validate and protect the digital copy of Holy Quran by employing a fragile watermark [13]. The Host image is decomposed using $2^{nd}$ and $4^{th}$ discrete wavelet transform DWT and the watermark image is inserted into its coefficients. This method is evaluated depending on small values of embedding factor K and when K is increased, the values of PSNR and SSIM decreased significantly. Also, the embedding procedure doesn't provide a sufficient level of security to the watermark.

## III. PROPOSED METHODOLOGY

Our work focuses on the authentication of the copies of the Holy Quran that are circulating among Muslims through the Internet or smart devices and discover any alteration. Quran images are used to reach that goal through applying image watermarking technique in transform domain where DWT is assigned to decompose the Quran cover image. The watermark is specified by scrambling the cover image itself. This in turn contributes to reduce the required data during the watermark extraction when there is any doubt. As Quran pages are nearly 604 one, so using a blind algorithm is more flexible to reach the desired goal.

### *A. Image scrambling*

Digital image scrambling is a method to encrypt the image content. Its purpose is to disguise the image to become difficult for intruders to be distinguished. So the original image is converted by scrambling into random format which is meaningless and unrecognizable by users [14].

Like other methods, our procedure split into two parts. The first step is concerning of embedding the specific created watermark into the host Quran image to get the watermarked Quran image. Then the second step comes to authenticate the watermarked distributed Quran image.

### *B. Embedding Procedure*

The embedding process of our proposed algorithm is illustrated in fig.1. The PNG cover image of digital Holy Quran pages has size of 512 x 368 and for the watermark image the size is 64 x46 or 32 x23. The host image and the watermark image are shown in fig. 2.

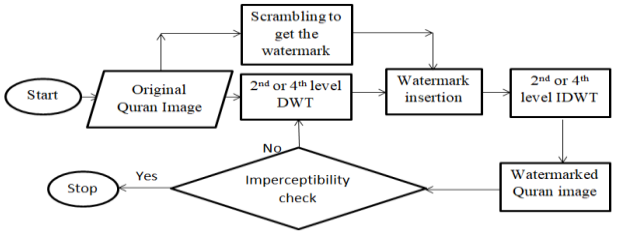
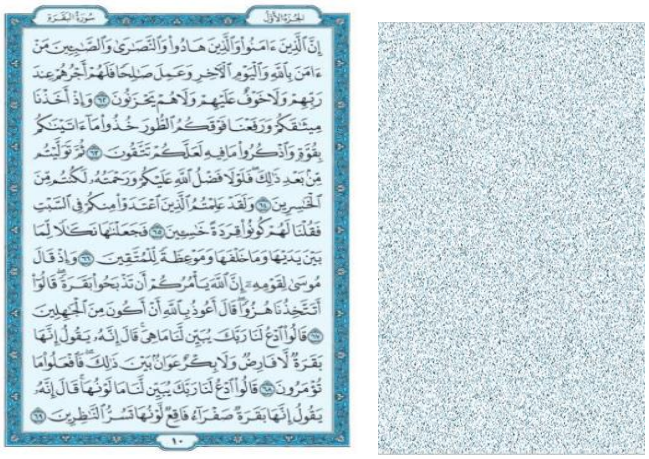



**Fig. 1. The embedding process.**

The watermark insertion steps are as follows:

1) Watermark generation and this is done by scrambling the host image. So each Quran page has its own unique watermark which generated by scrambling the Quran page image itself.



(a) (b)

**Fig. 2.(a)The original Quran image, (b)The watermark image.**

2) The cover Quran images are decomposed using DWT into 2nd (LL2, LH2, HL2, and HH2) and 4th (LL4, LH4, HL4, HH4) levels coefficients. This is accomplished by employing Haar filter Matlab tool and the output is shown in fig. 3 and 4.

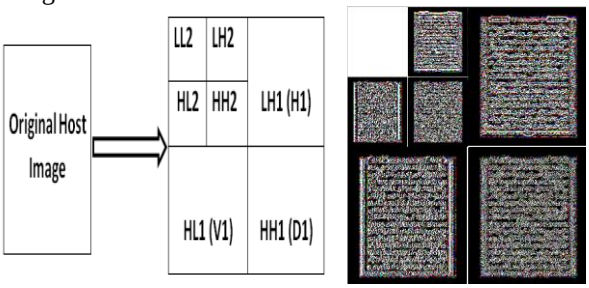
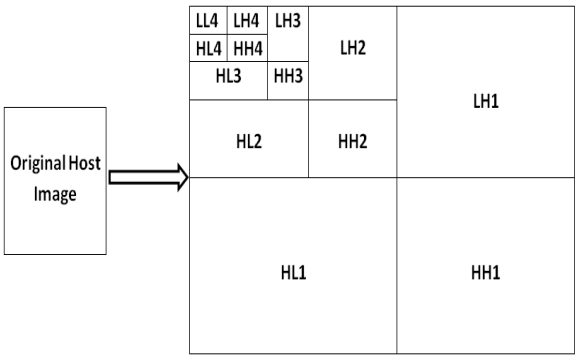



**Fig. 3.2nd level DWT.**



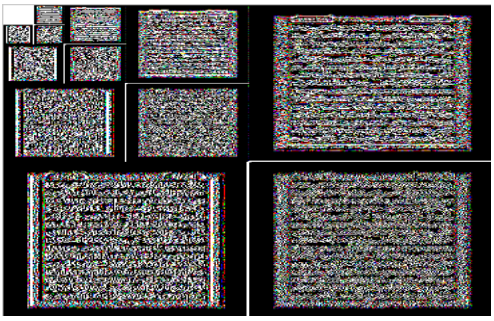



**Fig. 4.4th level DWT.**

3) The watermark image is inserted into the host Quran image coefficients using various values of embedding gain.
4) IDWT is performed to reconstruct the image and get the watermarked image.

Due to the sensitive nature of Holy Quran, the watermark should be unnoticeable. So as a confirmation on this issue, PSNR and MSE values are used as a measure of imperceptibility between the original image and the watermarked image. As image fidelity is a vital requirement for this application.

### *C. Extraction Procedure*

The process of detection and hence validating the Quran page image content is done by reversing the embedding steps. As our algorithm is blind so, there is no need for the existence of the cover image. The whole procedure is illustrated in detail in fig. 5. On this basis, document authentication is achieved according to the following steps:

1) The watermarked image is decomposed into its DWT 2nd (LL2, LH2, HL2, and HH2) or 4th (LL4, LH4, HL4, HH4) level coefficients and this is determined based on the level used in the embedding process.
2) Identifying the DWT coefficient in which the watermark was inserted and having the original watermark, the watermark is revealed from the required coefficient.
3) Then, IDWT is carried out to merge the image parts and the output is the extracted host image EHI.
4) The aforementioned original watermark is descrambled to produce the original host image OHI.
5) Finally OHI and EHI are compared together to decide whether the image is authentic or exposed to tamper or manipulation. This is achieved by identifying the error between the two versions. To confirm the validity of the image, the resulting error shouldn't exceed the value "1" and other than that value, the image is tampered.

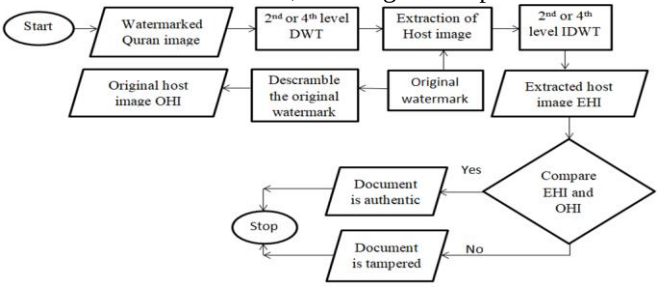



**Fig. 5.The authentication process.**

## IV. SYSTEM EVALUATION

### A. *Imperceptibility*

The main objective of this algorithm is to reveal any manipulation in the Quranic content by any saboteur. However, this is conditional on maintaining the image quality and integrity due to the paramount importance of the Holy Quran in the Islamic community. For this sake PSNR, SSIM and MSE are employed as a measure of imperceptibility evaluation between cover image and the resulting watermarked image [13].

The PSNR can be expressed in (1):

$$PSNR = 10\log_{10}\left(\frac{MAX^2}{MSE}\right) \quad (1)$$

Where PSNR is expressed in decibel (dB) and is defined as the ratio between the maximum value of the image pixels and the noise affecting that image [15].

Where MSE can be represented by (2):

$$MSE = \frac{1}{N}\sum_{m,n}(Q_0\,[m,n] - Q_w[m,n])^2 \quad (2)$$

Where:

N: The number of image pixels

$Q_0$ : Original host image,

$Q_w$: Watermarked image

And SSIM can be defined in (3):

$$SSIM(x,y) = [l(x,y)]^{\alpha}\,[c(x,y)]^{\beta}\,[s(x,y)]^{\gamma} \quad (3)$$

Where:

*l*: luminance component

*c*: contrast component

*s*: structure component

α, β and $\gamma$ are parameters to adjust the components

## V. RESULT AND DISCUSSION

The proposed algorithm uses two different levels of DWT to insert the identified watermark into the PNG format-stored host image. The first is the 2nd level DWT decomposition and the other one is the 4th level DWT decomposition.

### A. *Watermark Insertion in 2nd Level DWT Coefficients*

The sizes of cover image and watermark image are 512 x 368 and 64 x 46 respectively. The two datasets of embedding gain values (k) are (0.001, 0.002, 0.003, 0.004) and (2.1, 2.3, 2.5, 2.8, 3). The value of k should be carefully chosen as it is exploited to control the watermark invisibility. Increasing K value strengthens the watermarking algorithm and makes it more robust but on the other hand it degrades the quality of the watermarked image as it reduces the imperceptibility between the original and watermarked images. Here the watermark joined with k is inserted into the four DWT coefficients (LL2, LH2, HL2, HH2) separately i.e. at each time one coefficient is employed to include the mark. The values of PSNR, SSIM and MSE for each region at the different embedding gain are listed in table I and table II respectively.

The results in table I and II indicate that for our algorithm the four host regions have almost convergent results of PSNR values and as the value of k increases, the value of PSNR decreases. Also SSIM values decrease with increasing k, while MSE value goes up. According to these values, it's found that LL2 is the most appropriate region to hide a watermark, keeping in mind that other regions are also efficient.

**Table-I: PSNR (dB) for 2-level DWT Decomposition with No attack**

| K | A2(LL2) | | V2(LH2) | | H2(HL2) | | D2(HH2) | |
|---|---|---|---|---|---|---|---|---|
| | ***PSNR*** | | ***PSNR*** | | ***PSNR*** | | ***PSNR*** | |
| | ours | Olanrewaju's | ours | Olanrewaju's | ours | Olanrewaju's | ours | Olanrewaju's |
| **0.001** | INF | 75.3453 | INF | 75.3453 | INF | 75.3453 | INF | 75.3453 |
| **0.002** | INF | 69.3247 | INF | 69.3247 | INF | 69.3247 | INF | 69.3247 |
| **0.003** | INF | 65.8029 | INF | 65.8029 | INF | 65.8029 | INF | 65.8029 |
| **0.004** | INF | 63.3041 | INF | 63.3041 | INF | 63.3041 | INF | 63.3041 |
| | …… | …….. | …….. | …… | …… | …….. | …… | …… |
| **2.1** | 87.6598 | 45.6937 | 88.8408 | 45.1793 | 88.8408 | 45.1340 | 86.2084 | 45.7399 |
| **2.3** | 52.5290 | 45.6664 | 51.6349 | 45.0213 | 51.6292 | 44.9197 | 51.6247 | 45.6116 |
| **2.5** | 49.9751 | 45.6378 | 49.1683 | 44.8742 | 49.1749 | 44.7960 | 49.1687 | 45.4848 |
| **2.8** | 49.6785 | 45.5099 | 48.8469 | 44.3772 | 48.8520 | 44.3149 | 48.8488 | 45.1604 |
| **3** | 49.6785 | 45.3554 | 48.8469 | 44.0692 | 48.8520 | 44.0525 | 48.8488 | 45.0382 |

**Table-II: SSIM and MSE for 2-level DWT Decomposition with No attack**

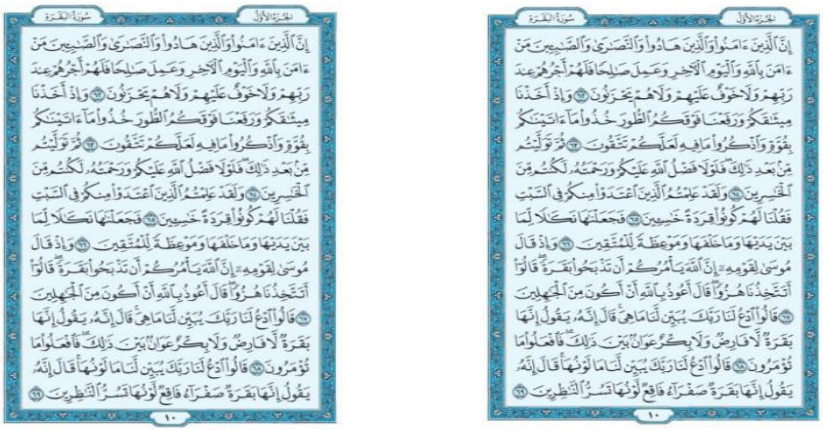
| K | A2(LL2) | | | | V2(LH2) | | | | H2(HL2) | | | | D2(HH2) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | |
| | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE |
| **0.001** | 1 | 0.0019 | 1 | 0 | 0.9998 | 0.0019 | 1 | 0 | 0.9999 | 0.0019 | 1 | 0 | 0.9998 | 0.0019 | 1 | 0 |
| **0.002** | 1 | 0.0076 | 1 | 0 | 0.9994 | 0.0076 | 1 | 0 | 0.9994 | 0.0076 | 1 | 0 | 0.9994 | 0.0076 | 1 | 0 |
| **0.003** | 1 | 0.0171 | 1 | 0 | 0.9989 | 0.0171 | 1 | 0 | 0.9988 | 0.0171 | 1 | 0 | 0.9988 | 0.0171 | 1 | 0 |
| **0.004** | 1 | 0.0304 | 1 | 0 | 0.9982 | 0.0304 | 1 | 0 | 0.9981 | 0.0304 | 1 | 0 | 0.9982 | 0.0304 | 1 | 0 |
| …… | …….. | …….. | …….. | …… | …….. | …….. | …….. | …… | …….. | …… | …….. | …… | …….. | …….. | …….. | …… |
| **2.1** | 0.9983 | 0.5029 | 1 | 0 | 0.9979 | 0.5836 | 1 | 0 | 0.9979 | 0.5864 | 1 | 0 | 0.9983 | 0.5005 | 1 | 0 |
| **2.3** | 0..9982 | 0.4995 | 0.9998 | 0.0034 | 0.9978 | 0.6125 | 0.9988 | 0.0043 | 0.9977 | 0.6142 | 0.9988 | 0.0048 | 0.9982 | 0.5147 | 0.9988 | 0.0048 |
| **2.5** | 0.9982 | 0.5017 | 0.9998 | 0.1919 | 0.9976 | 0.6360 | 0.9981 | 0.2247 | 0.9976 | 0.6349 | 0.9981 | 0.2241 | 0.9981 | 0.5264 | 0.9981 | 0.2248 |
| **2.8** | 0.9980 | 0.5467 | 0.9999 | 0.2382 | 0.9971 | 0.7242 | 0.9980 | 0.2851 | 0.9970 | 0.7199 | 0.9980 | 0.2847 | 0.9979 | 0.5764 | 0.9980 | 0.2849 |
| **3** | 0.9978 | 0.5765 | 0.9999 | 0.2382 | 0.9976 | 0.7825 | 0.9980 | 0.2851 | 0.9976 | 0.7719 | 0.9980 | 0.2847 | 0.9978 | 0.5988 | 0.9980 | 0.2849 |

### *B. Watermark Insertion in 4th Level DWT Coefficients*

For 4th level DWT decomposition, the cover image size is 512 x 368 and the size of the watermark image is 64 x 46. The two datasets of embedding factor values (k) are (0.001, 0.002, 0.003, 0.004) and (8.7, 8.9, 9.1, 9.3, 9.5). Here the watermark joined with k is inserted into the four DWT coefficients (LL4, LH4, HL4, HH4) separately i.e. at each time one coefficient is employed to include the watermark. The values of PSNR, SSIM and MSE for each region at the different embedding factor are listed in table III and IV respectively. From the table, the values of the PSNR, MSE and SSIM are the same in the four embedding regions for k values of 0.001, 0.002, 0.003 and 0.004. Increasing the range of embedding factor to 8.7, HH4 region has a higher value of PSNR than other regions, while the values of MSE and SSIM are the same for all regions. With increasing the gain farther, LL4 region has the best values of these parameters.

### *C. Comparison between 2nd and 4th Level DWT*

Comparing the results of 2nd level and 4th level of DWT decomposition, it's found that 4th level has higher values of PSNR, MSE and SSIM. This brings us to the concept that the resulting watermarked image by 4th level DWT decomposition is more imperceptible than that one obtained by 2nd level DWT decomposition. And both cases provide an excellent level of imperceptibility and this is obvious in fig. 6 and fig. 7.



(a) (b)

**Fig. 6. (a) LL2 watermarked image with k=2.1, (b) Extracted image.**

(a) (b)

**Fig. 7. (a) LL4 watermarked image with k=2.1, (b) Extracted image.**

**Table-III: PSNR (dB) for 4-level DWT Decomposition with No attack**

| K | A4(LL4) | | V4(LH4) | | H4(HL4) | | D4(HH4) | |
|---|---|---|---|---|---|---|---|---|
| | PSNR | | PSNR | | PSNR | | PSNR | |
| | ours | Olanrewaju's | ours | Olanrewaju's | ours | Olanrewaju's | ours | Olanrewaju's |
| **0.001** | INF | 87.7088 | INF | 87.7088 | INF | 87.7088 | INF | 87.7088 |
| **0.002** | INF | 81.6882 | INF | 81.6882 | INF | 81.6882 | INF | 81.6882 |
| **0.003** | INF | 78.1664 | INF | 78.1664 | INF | 78.1664 | INF | 78.1664 |
| **0.004** | INF | 75.6676 | INF | 75.6676 | INF | 75.6676 | INF | 75.6676 |
| ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... | ..... |
| **8.7** | 78.0189 | 53.8820 | 73.6420 | 53.5181 | 80.0184 | 53.5212 | 81.5878 | 53.5234 |
| **8.9** | 58.9260 | 53.4951 | 58.4425 | 53.1115 | 58.0414 | 53.1163 | 58.4301 | 53.1176 |
| **9.1** | 53.4669 | 53.1344 | 52.5848 | 52.7330 | 52.6363 | 52.7359 | 52.5061 | 52.7363 |
| **9.3** | 51.6657 | 52.7558 | 50.7416 | 52.3369 | 50.7855 | 52.3401 | 50.8147 | 52.3392 |
| **9.5** | 51.4111 | 52.4141 | 50.5641 | 51.9776 | 50.5780 | 51.9787 | 50.5519 | 51.9771 |

**Table-IV: SSIM and MSE for 4-level DWT Decomposition with No attack**

| K | A4(LL4) | | | | V4(LH4) | | | | H4(HL4) | | | | D4(HH4) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | |
| | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE | SSIM | MSE |
| **0.001** | 1 | 0.0001 | 1 | 0 | 1 | 0.0001 | 1 | 0 | 1 | 0.0001 | 1 | 0 | 1 | 0.0001 | 1 | 0 |
| **0.002** | 1 | 0.0004 | 1 | 0 | 1 | 0.0004 | 1 | 0 | 1 | 0.0004 | 1 | 0 | 1 | 0.0004 | 1 | 0 |
| **0.003** | 1 | 0.001 | 1 | 0 | 0.9999 | 0.001 | 1 | 0 | 0.9999 | 0.001 | 1 | 0 | 0.9999 | 0.001 | 1 | 0 |
| **0.004** | 1 | 0.0018 | 1 | 0 | 0.9999 | 0.0018 | 1 | 0 | 0.9999 | 0.0018 | 1 | 0 | 0.9999 | 0.0018 | 1 | 0 |
| …… | …….. | | …….. | …… | …… | …….. | …… | | …… | …… | …… | | | | | |
| **8.7** | 0.9998 | 0.0592 | 1 | 0 | 0.9997 | 0.0642 | 1 | 0 | 0.9997 | 0.0640 | 1 | 0 | 0.9996 | 0.0641 | 1 | 0 |
| **8.9** | 0.9998 | 0.0636 | 0.9999 | 0 | 0.9997 | 0.0688 | 0.9998 | 0 | 0.9997 | 0.0686 | 0.9997 | 0 | 0.9996 | 0.0687 | 0.9997 | 0 |
| **9.1** | 0.9998 | 0.0671 | 0.9998 | 0 | 0.9996 | 0.0727 | 0.9992 | 0 | 0.9996 | 0.0725 | 0.9992 | 0 | 0.9995 | 0.0726 | 0.9991 | 0 |
| **9.3** | 0.9998 | 0.0696 | 0.9998 | 0 | 0.9996 | 0.0754 | 0.9991 | 0 | 0.9996 | 0.0752 | 0.9991 | 0 | 0.9995 | 0.0753 | 0.9989 | 0 |
| **9.5** | 0.9997 | 0.0721 | 0.9998 | 0.0078 | 0.9995 | 0.0783 | 0.9991 | 0.0093 | 0.9995 | 0.0780 | 0.9991 | 0.0085 | 0.9994 | 0.0782 | 0.9988 | 0.0098 |

### *D. Effect of Salt and Pepper Attack*

The watermarked Quran image was exposed to salt and pepper noise with 0.02 noise density. Table V and VI illustrate the values of PSNR and SSIM when salt and pepper attack is applied on watermarked images using 2nd and 4th level DWT decomposition. By comparing the obtained results of PSNR and SSIM of attacked image in the table V and VI with the results of PSNR and SSIM without applied attack, it is noticeable that their values are significantly decreased in all cases. The lower the values of PSNR and SSIM of a tampered image, the greater the fragility of the scheme to attacks and this is the desired goal of the proposed algorithm.

### *E. Effect of Gaussian noise*

Gaussian noise with variance of 0.02 is applied to the watermarked image. The PSNR and SSIM values in tables VII and VIII obtained when the watermarked image is exposed to Gaussian noise for both 2nd and 4th level DWT is much lower than its counterpart when there is no attack. This, of course, shows the ability of the algorithm to detect any manipulations. Fig. 8 and fig. 9 illustrate the watermarked image after salt and pepper and Gaussian noise attacks respectively.

The two aforementioned attacks tend to remove the watermark by altering the watermarked image features and manipulating the matrices values that representing the image.

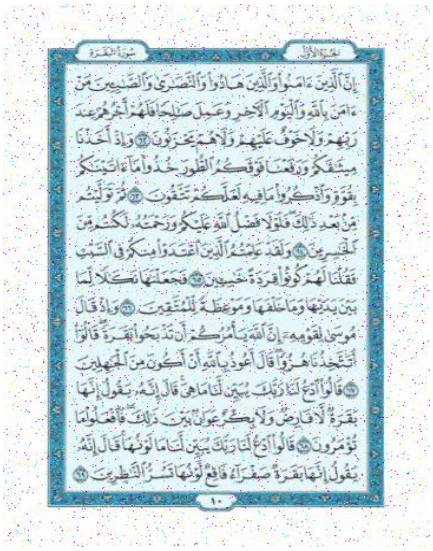


**Fig. 8.Watermarked image after salt and pepper attack.**

**Fig. 9.Watermarked image after Gaussian Noise attack.**

**Table-V: PSNR (dB) and SSIM under Salt & Pepper Attack for 2-level DWT Decomposition**

| K | A2(LL2) | | | | V2(LH2) | | | | H2(HL2) | | | | D2(HH2) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | |
| | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
| **0.001** | 20.7703 | 0.7559 | 20.7345 | 0.6621 | 20.7571 | 0.7521 | 20.7543 | 0.6668 | 20.7828 | 0.7555 | 20.7534 | 0.6656 | 20.6945 | 0.7511 | 20.6599 | 0.6584 |
| **0.002** | 20.7315 | 0.7524 | 20.6566 | 0.6595 | 20.7830 | 0.7576 | 20.6614 | 0.6578 | 20.7438 | 0.7544 | 20.8109 | 0.6669 | 20.7917 | 0.7539 | 20.6690 | 0.6604 |
| **0.003** | 20.7297 | 0.7516 | 20.7404 | 0.6619 | 20.7454 | 0.7516 | 20.6409 | 0.6585 | 20.7857 | 0.7563 | 20.6463 | 0.6570 | 20.7984 | 0.7542 | 20.6967 | 0.6611 |
| **0.004** | 20.7662 | 0.7554 | 20.6710 | 0.6622 | 20.7425 | 0.7549 | 20.8620 | 0.6703 | 20.7453 | 0.7520 | 20.6666 | 0.6615 | 20.7337 | 0.7531 | 20.7067 | 0.6619 |
| …… | …….. | …….. | …….. | …… | …….. | …….. | …….. | …….. | …….. | …….. | …….. | …….. | …….. | …….. | …….. | …….. |
| **2.1** | 20.7164 | 0.7504 | 20.7460 | 0.6633 | 20.7977 | 0.7571 | 20.8064 | 0.6664 | 20.7681 | 0.7535 | 20.6928 | 0.6605 | 20.7728 | 0.7531 | 20.7040 | 0.6632 |
| **2.3** | 20.6633 | 0.7492 | 20.7989 | 0.6663 | 20.7376 | 0.7545 | 20.7829 | 0.6668 | 20.7297 | 0.7528 | 20.7439 | 0.6629 | 20.7103 | 0.7518 | 20.7142 | 0.6620 |
| **2.5** | 20.7107 | 0.7529 | 20.6767 | 0.6573 | 20.7570 | 0.7555 | 20.6391 | 0.6560 | 20.7384 | 0.7509 | 20.7450 | 0.6646 | 20.7146 | 0.7521 | 20.7175 | 0.6625 |
| **2.8** | 20.6195 | 0.7473 | 20.6593 | 0.6616 | 20.7605 | 0.7540 | 20.6608 | 0.6592 | 20.7304 | 0.7510 | 20.7376 | 0.6644 | 20.7242 | 0.7531 | 20.7844 | 0.6649 |
| **3** | 20.7391 | 0.7530 | 20.7337 | 0.6641 | 20.7470 | 0.7531 | 20.7574 | 0.6678 | 20.7471 | 0.7522 | 20.7773 | 0.6682 | 20.7707 | 0.7541 | 20.6096 | 0.6549 |

**Table-VI: PSNR (dB) and SSIM under Salt & Pepper Attack for 4-level DWT Decomposition**

| K | A4(LL4) | | | | V4(LH4) | | | | H4(HL4) | | | | D4(HH4) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | |
| | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
| **0.001** | 20.7602 | 0.7531 | 20.7934 | 0.6658 | 20.6802 | 0.7505 | 20.6997 | 0.6595 | 20.7243 | 0.7517 | 20.7838 | 0.6656 | 20.7735 | 0.7557 | 20.7637 | 0.6637 |
| **0.002** | 20.7681 | 0.7533 | 20.7460 | 0.6644 | 20.7603 | 0.7512 | 20.7665 | 0.6647 | 20.8175 | 0.7579 | 20.8010 | 0.6665 | 20.7728 | 0.7550 | 20.7624 | 0.6672 |
| **0.003** | 20.6639 | 0.7519 | 20.8003 | 0.6667 | 20.7127 | 0.7519 | 20.7219 | 0.6605 | 20.7123 | 0.7509 | 20.7061 | 0.6612 | 20.7415 | 0.7522 | 20.6546 | 0.6607 |
| **0.004** | 20.7571 | 0.7534 | 20.6356 | 0.6602 | 20.6685 | 0.7506 | 20.6973 | 0.6620 | 20.7304 | 0.7520 | 20.7727 | 0.6635 | 20.6699 | 0.7499 | 20.7133 | 0.6619 |
| ……. | ……. | | ……. | | ……. | | ……. | | ……. | | ……. | | ……. | | ……. | |
| **8.7** | 20.7103 | 0.7503 | 20.7134 | 0.6626 | 20.7702 | 0.7518 | 20.6860 | 0.6601 | 20.6664 | 0.7510 | 20.6541 | 0.6611 | 20.7504 | 0.7552 | 20.8274 | 0.6675 |
| **8.9** | 20.7308 | 0.7525 | 20.7754 | 0.6649 | 20.7827 | 0.7543 | 20.7124 | 0.6626 | 20.6999 | 0.7510 | 20.7265 | 0.6645 | 20.7262 | 0.7510 | 20.6662 | 0.6590 |
| **9.1** | 20.8345 | 0.7546 | 20.8465 | 0.6735 | 20.6851 | 0.7503 | 20.6637 | 0.6614 | 20.7446 | 0.7528 | 20.7296 | 0.6634 | 20.7565 | 0.7526 | 20.7691 | 0.6661 |
| **9.3** | 20.6684 | 0.7489 | 20.7174 | 0.6608 | 20.7130 | 0.7530 | 20.6503 | 0.6567 | 20.7843 | 0.7546 | 20.6878 | 0.6589 | 20.7593 | 0.7543 | 20.7939 | 0.6663 |
| **9.5** | 20.7369 | 0.7521 | 20.7316 | 0.6658 | 20.7660 | 0.7537 | 20.7548 | 0.6630 | 20.7209 | 0.7499 | 20.6637 | 0.6615 | 20.7507 | 0.7540 | 20.6648 | 0.6591 |

**Table-VII: PSNR (dB) and SSIM under Gaussian noise for 2-level DWT Decomposition**

| K | A2(LL2) | | | | V2(LH2) | | | | H2(HL2) | | | | D2(HH2) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | | *Olanrewaju's* | | *Ours* | |
| | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
| **0.001** | 22.1125 | 0.7443 | 22.2628 | 0.6309 | 22.0856 | 0.7429 | 22.2741 | 0.6314 | 22.0958 | 0.7438 | 22.2628 | 0.6306 | 22.0956 | 0.7441 | 22.2689 | 0.6305 |
| **0.002** | 22.1173 | 0.7443 | 22.2515 | 0.6307 | 22.1066 | 0.7436 | 22.2604 | 0.6303 | 22.1035 | 0.7437 | 22.2542 | 0.6306 | 22.0945 | 0.7437 | 22.2609 | 0.6309 |
| **0.003** | 22.1048 | 0.7439 | 22.2590 | 0.6300 | 22.0847 | 0.7431 | 22.2595 | 0.6301 | 22.0914 | 0.7434 | 22.2401 | 0.6291 | 22.1046 | 0.7437 | 22.2460 | 0.6305 |
| **0.004** | 22.1055 | 0.7440 | 22.2483 | 0.6307 | 22.1054 | 0.7435 | 22.2571 | 0.6305 | 22.0983 | 0.7432 | 22.2433 | 0.6300 | 22.0894 | 0.7437 | 22.2599 | 0.6303 |
| …… | …….. | …….. | …… | …… | …….. | …….. | …… | …… | …….. | …….. | …… | …… | …….. | …….. | …… | …… |
| **2.1** | 22.0900 | 0.7436 | 22.2535 | 0.6304 | 22.0989 | 0.7435 | 22.2563 | 0.6304 | 22.0802 | 0.7429 | 22.2574 | 0.6304 | 22.1153 | 0.7447 | 22.2530 | 0.6305 |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **2.3** | 22.0969 | 0.7442 | 22.2746 | 0.6320 | 22.0881 | 0.7434 | 22.2341 | 0.6299 | 22.0844 | 0.7432 | 22.2424 | 0.6297 | 22.0878 | 0.7434 | 22.2367 | 0.6289 |
| **2.5** | 22.1143 | 0.7447 | 22.2865 | 0.6319 | 22.0808 | 0.7433 | 22.2290 | 0.6285 | 22.0933 | 0.7435 | 22.2121 | 0.6284 | 22.0852 | 0.7427 | 22.2407 | 0.6286 |
| **2.8** | 22.0868 | 0.7435 | 22.2930 | 0.6318 | 22.0822 | 0.7435 | 22.2230 | 0.6289 | 22.0695 | 0.7428 | 22.2107 | 0.6280 | 22.0805 | 0.7431 | 22.2240 | 0.6289 |
| **3** | 22.1104 | 0.7437 | 22.2699 | 0.6313 | 22.0734 | 0.7429 | 22.2147 | 0.6283 | 22.0740 | 0.7429 | 22.2170 | 0.6283 | 22.0830 | 0.7431 | 22.2185 | 0.6286 |

**Table-VIII: PSNR (dB) and SSIM under Gaussian noise for 4-level DWT Decomposition**

| K | A4(LL4) | | | | V4(LH4) | | | | H4(HL4) | | | | D4(HH4) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | | Olanrewaju's | | Ours | |
| | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
| **0.001** | 22.0987 | 0.7439 | 22.2575 | 0.6310 | 22.0925 | 0.7431 | 22.2700 | 0.6315 | 22.1031 | 0.7437 | 22.2631 | 0.6312 | 22.0902 | 0.7437 | 22.2594 | 0.6308 |
| **0.002** | 22.0989 | 0.7435 | 22.2346 | 0.6297 | 22.0921 | 0.7435 | 22.2745 | 0.6311 | 22.1038 | 0.7442 | 22.2689 | 0.6310 | 22.0943 | 0.7433 | 22.2571 | 0.6296 |
| **0.003** | 22.0942 | 0.7435 | 22.2552 | 0.6309 | 22.0838 | 0.7431 | 22.2270 | 0.6299 | 22.0924 | 0.7435 | 22.2519 | 0.6302 | 22.1047 | 0.7440 | 22.2628 | 0.6302 |
| **0.004** | 22.0977 | 0.7441 | 22.2744 | 0.6317 | 22.1116 | 0.7441 | 22.2548 | 0.6304 | 22.0921 | 0.7435 | 22.2554 | 0.6302 | 22.0880 | 0.7431 | 22.2667 | 0.6307 |
| …… | …… | | …… | | …… | | …… | | …… | | …… | | …… | | …… | |
| **8.7** | 22.0860 | 0.7432 | 22.2622 | 0.6306 | 22.0839 | 0.7434 | 22.2551 | 0.6306 | 22.0920 | 0.7441 | 22.2457 | 0.6298 | 22.0842 | 0.7431 | 22.2356 | 0.629 |
| **8.9** | 22.0992 | 0.7434 | 22.2715 | 0.6317 | 22.0926 | 0.7436 | 22.2560 | 0.6306 | 22.0962 | 0.7433 | 22.2517 | 0.6303 | 22.0751 | 0.7425 | 22.2639 | 0.6308 |
| **9.1** | 22.0951 | 0.7440 | 22.2692 | 0.6319 | 22.0869 | 0.7428 | 22.2358 | 0.6299 | 22.0868 | 0.7432 | 22.2422 | 0.6301 | 22.0885 | 0.7434 | 22.2499 | 0.6301 |
| **9.3** | 22.0940 | 0.7434 | 22.2901 | 0.6329 | 22.0841 | 0.7426 | 22.2303 | 0.6290 | 22.0828 | 0.7426 | 22.2355 | 0.6300 | 22.0742 | 0.7428 | 22.2290 | 0.6294 |
| **9.5** | 22.0947 | 0.7439 | 22.2614 | 0.6320 | 22.0986 | 0.7436 | 22.2382 | 0.6296 | 22.0865 | 0.7432 | 22.2195 | 0.6285 | 22.0793 | 0.7435 | 22.2434 | 0.6293 |

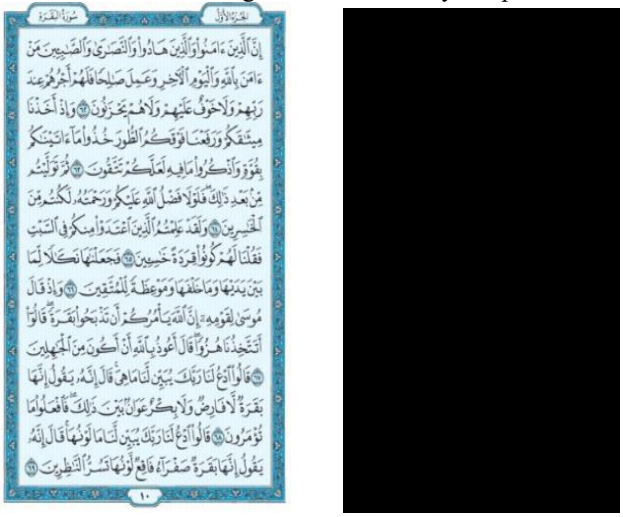
### *F. Effect of Cropping*

This type of geometrical attacks aims to distort certain portions of the watermarked image to adversely affect the watermark extraction process. Experimentally the attack is implemented by removing some parts from the watermarked image. The watermarked image is displayed in fig. 10 after deleting small parts of it. It's found that the values of PSNR and SSIM significantly decreased to 14.3302 dB and 0.5199 respectively.

**Fig. 10. Watermarked image after cropping attack.**

### G. Tamper Localization

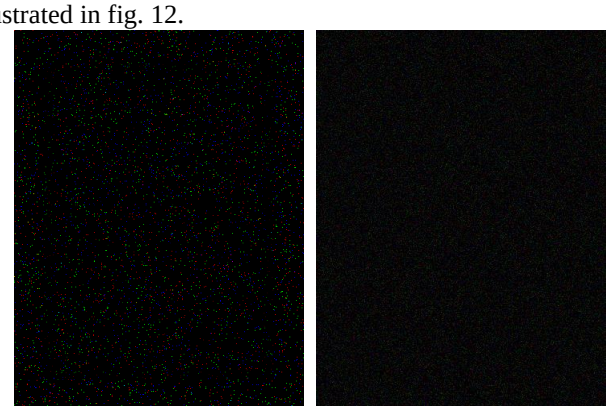
The proposed algorithm is evaluated based on watermarked image quality using PSNR, SSIM and MSE. It's important for our scheme to have a localization capability. Tamper location is identified by calculating the difference in pixel values between the original image and the watermarked one. Thus the image of the difference becomes an indication of where the tamper occurred. For the original image in Fig. 2 (a) and the watermarked image with no tamper in Fig. 11 (a), the difference image between the is shown in Fig. 11 (b). From the difference image, it's clear that the watermarked image doesn't have any tamper.



(a) (b)

**Fig. 11. (a) Watermarked image, (b) Difference image.**

But with concerning to the watermarked images in fig. 8 and fig. 9 which are exposed to salt and pepper attack and Gaussian noise respectively, the difference image is illustrated in fig. 12.



**(a) (b)**

**Fig. 12. Difference image for (a) Salt and pepper, (b) Gaussian noise attacked watermarked images.**

### H. Comparison with Related Work

Regarding to the results of PSNR, SSIM and MSE in tables 4.1, 4.2, 5.1 and 5.2, it's found that for all embedding regions, the proposed method is superior to Olanrewaju's algorithm and this led to the advantage of having an excellent level of imperceptibility. This was done at the expense of increasing the complexity of our scheme by adding the scrambling step.

Also with comparing the results when attacks are added, we observe that the values of PSNR achieved by our algorithm are a little bit higher than the values obtained by Olanrewaju [13]. Whereas our SSIM value is lower than the one in Olanrewaju's algorithm. These results lead us to the fact that our algorithm is more fragile than the method presented by Olanrewaju.

## VI. CONCLUSION AND FUTURE DIRECTIONS

For Holy Quran text images authentication purpose, a blind fragile image watermarking scheme is presented in this paper. DWT in addition to scrambling principle are employed to achieve this goal. The scrambled watermark strengthens the performance of the algorithm. Coefficients of both 2nd and 4th levels of DWT are exploited to conceal the watermark at different levels of embedding factor K. Simulation results proved that the four regions of 2nd level of DWT gives convergent values of PSNR, SSIM and MSE for a specific K value and so does the 4th level of DWT coefficients. The proposed method has achieved a better level of imperceptibility than other methods. Besides, watermarking using 4th level of DWT decomposition is more imperceptible than that obtained by 2nd level DWT decomposition. After being attacked, the results obtained using the watermarked attacked image verifies that the watermark is damaged. So the fragility characteristic is successfully utilized. This algorithm is also applicable to traditional color images and achieves the same performance. In future research, we will work to find new, more effective ways to correct the tampered part and restore the cropped section.

## REFERENCES

1. X. Zhou, W. Zhao, Z. Wang and L. Pan, "Security theory and attack analysis for text watermarking", Proceedings of the 2009 International Conference on E-Business and Information System Security, May 23-24, Wuhan, IEEE, (2009), pp. 1-6.
2. F. Kurniawan, M. S. Khalil, M. K. Khan and Y. M. Alginahi," Exploiting digital watermarking to preserve Integrity of the digital Holy Quran images", in IEEE 2013 Taibah University International Conference on Advances in Information Technology for the Holy Quran and Its Sciences, (2013), pp. 30-36.
3. X. Zhou , H. Zhang and C. Wang, "A Robust image watermarking technique based on DWT, APDCBT, and SVD" , Journal symmetry, (2018), pp. 1-14.
4. H. Zhang, C. Wang and X. Zhou, "A Robust Image Watermarking Scheme Based on SVD in the Spatial Domain", Journal of Future Internet, (2017), pp. 1-16.
5. M. Abdullatif, A. M. Zeki, J. Chebil, and T. S. Gunawan, "Properties of digital image watermarking," in 2013 IEEE 9th International Colloquium on Signal Processing and its Applications, (2013), pp. 235–240.
6. M. A. AlAhmad, I. Alshaikhli, and A. E. Alduwaikh, "A new fragile digital watermarking technique for a PDF digital Holy Quran," in 2013 International Conference on Advanced Computer Science Applications and Technologies, 2013, pp. 250–253.
7. F. Kurniawan, M. S. Khalil, M. K. Khan and Y. M. Alginahi," DWT+LSB-based fragile watermarking method for digital Quran images", in IEEE 2014 International Symposium on Biometrics and Security Technologies (ISBAST), (2014), pp. 290-297.
8. R. K. MOVAGHAR and H. K. BIZAKI," A new approach for digital image watermarking to predict optimal blocks using artificial neural networks", Turkish Journal of Electrical Engineering & Computer Sciences, (2017), pp. 644-654.
9. A. K. Singh," Improved hybrid algorithm for robust and imperceptible multiple watermarking using digital images", Springer Science+Business Media New York, (2016), pp. 1-18.
10. Almazrooie, M., et al. "Integrity verification for digital Holy Quran verses using cryptographic hash function and compression". Journal of King Saud University – Computer and Information Sciences (2018), pp. 1-11.

11. R.A. Alotaibi, L.A. Elrefaei,"Text-image watermarking based on integer wavelet transform (IWT) and discrete cosine transform (DCT)", Applied Computing and Informatics (2018), pp. 1-12.
12. M. S. Khalil, F. Kurniawan, M. K. Khan and Y. M. Alginahi," Two-Layer fragile watermarking method secured with chaotic map for authentication of digital Holy Quran", Scientific World Journal, (2013), pp. 1-29.
13. R. Olanrewaju, F. Fajingbesi and N. Ishak, "Watermarking in protecting and validating the integrity of digital information: A case study of the Holy Scripture", proceedings of the 2016 6th International Conference on Information and Communication Technology for The Muslim World., IEEE, (2016), pp. 222-227.
14. J. Dong, G. Wu, T. Yang and Y. Li," The improved image scrambling algorithm for the wireless image transmission systems of UAVs", Sensors Journal, (2018), pp. 1-16.
15. M. Abdullatif, O. O. Khalifa, R. F. Olanrewaju, and A. M. Zeki, "Robust image watermarking scheme by discrete wavelet transform," in 2014 International Conference on Computer and Communication Engineering, September (2014), pp. 316–319.


## AUTHORS PROFILE

**Amira Eid** was born in Menya Al-kamh, Al-sharkia governorate, Egypt in 1985. She got BSc degree in electronic engineering and electrical communications, Menouf, Menoufya University in 2007. She is a master student in department of electrical engineering, Al- Azhar University. She is interested in researching in wireless communications and digital signal processing. She has a publication in the field of text watermarking titled "A Tamper proofing Text Watermarking Shift Algorithm for Copyright Protection". She is working as a teaching assistant in electrical engineering department in HTI, 10th of Ramadan city, Egypt. She is interested in teaching the fields of Electrical circuits, Electronics and wireless communications.

**Ahmed A. Emran** was born in Cairo in 1983. He got the bachelor and master from Al-Azhar university in 2006 and 2011 respectively. He got the PhD from Egypt Japan University of Science and Technology (EJUST), Alexandria, Egypt in 2015 in Electronics and Communication Engineering. Thesis title is Performance Enhancement of LDPC Codes. He has presented several publications in the field of communication engineering. He is interested in researching in wireless communications, forward error correction and information theory. He is an assistant professor in faculty of engineering, Al-Azhar University. He is interested in teaching the fields of Electrical circuits, field theorem, transmission line theorem, antenna theorem, microwave devices and communication networks.

**Ahmed Yahya** is a Professor of Electronics, working in the Department of Electrical Engineering, Al-Azhar University, Nasr City, Cairo-11371, Egypt. He received his bachelor and master degrees in electrical engineering from Al-Azhar University. He received an electrical engineering Ph.D. in 1998 from Ain Shams University, Integrated Circuit Lab, Cairo, Egypt. Prof. Ahmed Yahya has conducted major research projects in the development of microprocessors (high-performance, low-power, and mixed-signal). He has authored more than 200 peer-reviewed publications. His research interests are VLSI low power Devices, Mixed-Signal design CMOS Digital circuits, Timing models and design of voltage circuits, NB-IOT smart E-Health systems.