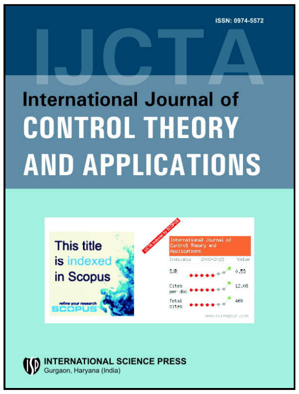


International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 15 • 2017

Experimental Analysis of DWT based Digital Image Watermarking

Ranjeet Kumar Singh[1], Manish Kumar[1] and Dilip Kumar Shaw[1]

[1] *Department of Computer Applications National Institute of Technology, Jamshedpur, Jharkhand,*
Emails: 2014rsca002@nitjsr.ac.in, manishkumar08353@nitjsr.ac.in, dkshaw.dca@nitjsr.ac.in

Abstract: Watermarking plays an important role in image security and authentication. In Image Watermarking some cover media is embedded within the original image to prevent it from unwanted user. A lot of algorithm is available for image security and authentication, Discrete Wavelet Transform (DWT) is one of them and it is mostly used in image processing. This paper explains different DWT based image watermarking scheme. In the first algorithm, DWT has been performed on Original image, and watermarking is done on Red, Green and Blue components of the Original image. This experiment is done at 1st level and 2nd level. In another experiment DWT has been applied on Original image and Watermark image at 1st level and chose low frequency sub-bands for Watermark embedding and extraction. The second section of this paper is impact of noise on Watermarked image. To measure the robustness of algorithms against noise, PSNR, SNR, MSE, CC and BCR have been calculated. A comparative analysis has been done for all the algorithms against different variance of noise.

Keywords: Discrete Wavelet Transform (DWT), PSNR, SNR, Correlation Coefficient

1. INTRODUCTION

In the era of internet, information is flowing from one place to another place quite rapidly. To protect the information from any misuse, other media is embedded within information is known as watermarking. In image watermarking, other media is embedded within image. The carrier media may be any one among the image, video, audio and text. Watermarked image should be robust enough against common image processing attacks like image blurring, sharpening, compression and cropping [1] [3]. Quality of watermarked image should be similar to the Original image. Watermarking systems have two steps: 1) Watermark embedding and 2) Watermark extracting. In Watermark embedding, the Original image embeds other media within it with the help of a key file. In the extraction process, Original image is retrieved with the help of the same key and the Watermarked image. The Watermarked images are of two types: Visible and Invisible. In visible watermarking, watermark is visible on the background of the original image. It is used mainly for the purpose of ownership and copyright protection. Invisible watermarking hides watermark within host image. Watermarking extraction techniques can be categorized into three categories: Non-blind, Semi-blind, and Blind. In Non-blind Watermarking extraction, secret key and original image is required.

In Semi-blind, watermark bit sequence and secret key is required for the extraction. Blind watermark can be extracted only with the secret key. Watermarking has a wide range of applications such as: Data authentication, Medical applications, Fingerprinting, Data hiding, Military applications and Copyright protection [2]. Digital watermarking techniques are mainly explored in two domains: transform domain and spatial domain. Spatial domain techniques mainly use the pixel locations for the watermarking. Least significant bits (LSB) and Intermediate Significant bits (ISB) are well known spatial domain techniques. In the transform domain the pixel values are transformed to the coefficients which are modified to embed watermark. Hence, Transform domain techniques are more robust than the spatial domain techniques. Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT) and Singular Value Decomposition (SVD) are mainly used in the transform domain techniques. Any watermarking technique should have some characteristics such as[1] [3]:

1) Quality: Quality of the image should not degrade after watermarking from human perception point of view.
2) Robustness: The watermarked data should be robust enough against common image processing attacks like filtering, compression, filtering with compression.
3) Capacity: The number of bits that a host image can have inside it at one time.

Most of the watermarking techniques follow either one or two of the above requirements and also not robust against all the common image processing attacks. So, developing a good watermarking technique is still a challenging task.

Rest of the paper is organized in 4 sections. Section II contains the literature survey of some related papers, section III describes about the Discrete Wavelet Transform technique and steps for Watermark embedding and Watermark extraction along with results and analysis. Paper is concluded in section IV.

2. LITERATURE SURVEY

Garcia et. al [4] proposed a watermarking algorithm based on DWT for self-recovery and authentication of tampered image. Daubechies DWT, Halftoning and QIM methods have been used to protect the digital image. IDWT, Inverse halftoning and median filtering have been used for authentication, detection and self-recovery of tampered regions. Watermarked images have PSNR values over 35dB; while Recovered images have PSNR values over 32 dB. Proposed algorithm shows good results against tamper detection and self-recovery of image.

Kamila et al. [5] presented a technique for image steganography using DWT. Image blocks are selected from sub-bands after applying DWT on the image.

Bedi et al. [6] used two levels DWT for image steganography. There is no need of any key file or original cover image for extraction purpose. MSE, PSNR, NC, NCC and NAE have been calculated to measure the performance of the algorithm.

Kadu et al. [7] proposed a method for video watermarking based on DWT. DWT has been applied on each frame of video to generate the key file for corresponding frame. Algorithm's robustness has been shown by calculating PSNR, NC and SSIM of original image and watermarked image.

Sharma et al. [8] devised a novel algorithm for audio compression using DWT. Size of the audio signal has been significantly reduced by removing zero coefficients. To analyze the performance of the algorithm, mathematical metrics SNR and distortion have been measured.

Selvi et al. [9] proposed a watermarking technique to hide the patient's information in medical images. Proposed method showed better results than existing methods on the basis of PSNR and MSE evaluation.

Wang et al. [10] proposed a digital watermarking technique in transform domain using DWT, DCT and SVD. They used the advantages of DWT, DCT and SVD to meet the requirements imperceptibility and robustness.

Arnold transform has been used to improve the robustness against the common image process like JPEG compression, noise, filtering, cutting, rotation and contrast. Proposed algorithm has been compared with the SVD and DCT+SVD, shows better results. Proposed algorithm with the arnold transform shows even more better results.

Nguyen et al. [11] presented a blind watermarking technique to improve the robustness and capacity. Proposed method is a combination of Discrete Wavelet Transform (DWT) and Singular Value Decomposition (SVD). In this method, appropriate SVs of HH sub-bands of the original image are replaced by SVs of the watermark. Key generation scheme ensures security to the watermark. Stirmark Benchmark 4.0 tool compares the result of the proposed method with other existing methods and showed better result against watermarking attacks.

Zhou et al. [12] applied two levels DWT to the carrier image before applying SVD on the selected sub-bands. Carrier image is decomposed into four sub-bands (LL1, LH1, HL1 and HH1) with the use of 2D DWT at the first level. In the 2nd level DWT is applied to the LH1 and HL1 of the first level and further decomposed them into eight sub-bands (LH1_LL2, LH1_LH2, LH1_HL2, LH1_HH2 and HL1_LL2, HL1_LH2, HL1_HL2, HL1_HH2). Among these eight sub-bands, four sub-bands (LH1_LH2, LH1_HL2, HL1_LL2 and HL1_LH2) are selected to apply the SVD. Inverse process has been applied for the watermark extraction purpose.

Jane et al. [13] presented a non-blind watermarking technique based on DWT and SVD. At first cover image is decomposed into four sub-bands (LL, LH, HL, HH) using DWT. SVD is applied to the LL sub-band of the decomposed image. A scaling factor is used to modify the diagonal singular value coefficients of LL sub-band along with watermark. SVD is again applied to reconstruct the LL sub-band coefficients. Finally watermarked image is obtained by computing inverse DWT. Peak signal-to-noise ratio (PSNR) and Normalized similarity ratio (NSR) have been calculated to measure the efficiency of the proposed algorithm.

For the literature, it is observed that majority of the studies on image watermarking have used algorithms based on transform domain techniques. Discrete Wavelet Transform has been used for audio, video and image watermarking in the literature. Literature also showed some hybrid techniques using DWT. DWT decomposes the image into vertically and horizontally low and high frequency components. One can choose region of interest of any digital image. Literature shows the importance of DWT. In this paper, we have shown an experimental analysis of 1 level DWT and 2 level DWT of color component based image watermarking technique, and other is 1 level DWT has been applied on Original but watermarking is not done on color component.

3. DWT BASED IMAGE WATERMARKING

3.1. Discrete Wavelet Transform

Discrete Wavelet Transform is a frequency domain technique which is mainly used in signal processing and image processing. This technique decomposes an image into sub-bands of low frequency components and high frequency components LL (approximation), LH (horizontal details), HL (Vertical details) and HH (Diagonal details). We can further subdivide any part of the image up to several levels using DWT. Most of the energy is concentrated within low frequency components. In this work DWT up to two levels have been shown. With the use of inverse DWT we can reconstruct the image [3].

3.2. Watermark Embedding and Extraction algorithm

3.2.1 Using Red, Green and Blue components of original image and watermark image after aplying 1 level DWT

Embedding Process

Step 1: Apply DWT to Original image

[LL1, HH1, HL1, HH1] = dwt2 (Original image);

Decomposed into 4 sub-bands: LL1, HH1, HL1 and HH1

Step 2: Find Red, Green and Blue Components of LL1 sub-band of Original image.

Img1 = LL1;

R1 = Img1 (:,:,1); G1 = Img1 (:,:,2); B1= Img1 (:,:,3);

Step 3: Apply DWT to Watermark image

[LLW1, LHW1, HLW1, HHW1] = dwt2 (Watermark image);

Decomposed into 4 sub-bands: LLW1, LHW1, HLW1 and HHW1

Step 4: Find Red, Green and Blue Components of LLW1 sub-band of Watermark image.

Img2 = LLW1;

R2 = Img2 (:,:,1); G2 = Img2 (:,:,2); B2 = Img2 (:,:,3);

Step 5: Embed the Red, Green and Blue components of Img2 to Img1 with factor 0.080.

R3 = R2 + (0.080*R1); G3 = G2 + (0.080*G1); B3 = B2 + (0.080*B1);

Step 6: Reconstruct LL1 components by combining new and modified Red, Green and Blue components.

Step 7: Use Inverse DWT to combine LL1, LH1, HL1 and HH1.

Recovery Process

Step 1: Apply DWT to Watermarked image.

[LLWED1, LHWED1, HLWED1, HHWED1] = dwt2 (Watermarked image);

Step 2: Find Red, Green and Blue components of LLWED1 sub-band of Original image.

Img3 = LLWED1;

R4 = Img3 (:,:,1); G4 = Img3 (:,:,2); B4 = Img3 (:,:,3);

Step 3: Apply DWT for Original image, to decompose into 4 sub-bands: LL1, LH1, HL1 and HH1.

[LL1, LH1, HL1, HH1] = dwt2 (Original image);

Step 4: Find Red, Green and Blue components of LL1 sub-band of Original image.

Img1 = LL1;

R5 = Img1 (:,:,1); G5 = Img1 (:,:,2); B5 = Img1 (:,:,3);

Step 5: Reconstruct the Red, Green and Blue components of LLW1 sub-band of Watermark image.

New Watermark Red = (Red component of LLWED1 - Red component of LL1)/0.080;

New Watermark Green = (Green component of LLWED1 - Green component of LL1)/0.080;

New Watermark Blue = (Blue component of LLWED1 - Blue component of LL1)/0.080;

Step 6: Reconstruct the LLW1 sub-band of Watermark image using modified Red, Green and Blue components.

Step 7: Combine the LLW1, LHW1, HLW1 and HHW1 sub-bands using Inverse DWT and get the *Recovered Watermark image.*

3.2.2. Results and Analysis

This experiment has been performed on MATLAB 7.0. In this experiment Original image and Watermark image has been shown in 1(a) and 1(b) respectively. At first, 1 level DWT has been applied on Original image and Watermark image. LL sub-band of Original image and Watermark image are combined to form Watermarked image. Watermarked image and Recovered Watermark image are shown in figure 1(c) and 1(d) respectively. Watermarked image and Recovered Watermark image are quite similar to the original image and Watermark image respectively.

(a) **(b)** **(c)** **(d)**

Figure 1: (a) Original image (b) Watermark image (c) Watermarked image (d) Recovered Watermark image

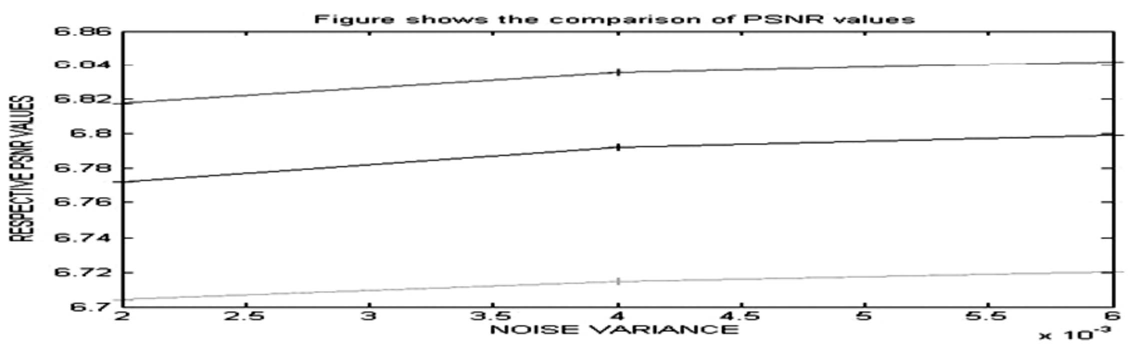
To measure the performance of the above experiments some metrices such as Avg. PSNR, SNR, MSE, CC and BCR have been calculated between Watermark and Recovered Watermark image on different noise with different variance. Table 1 shows the performance of the algorithm on gaussian noise with 0.002, 0.004 and 0.004. In Fig.2, Fig 4 and Fig 6, Red line represents Gaussian noise, Black line represents Speckle noise and Green line represents Salt and Pepper noise.

Table 1
Experiment Result

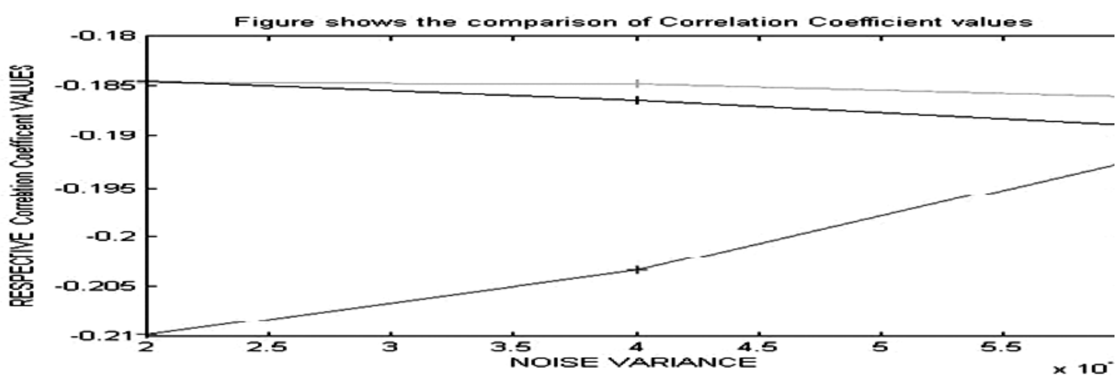
Gaussian	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
0.002	6.8108(R)	6.8174	-3.1613(R)	1.3552e+004(R)	-0.2098	0.2866
	6.7941(G)		-3.1779(G)	1.3604e+004(G)		
	6.8473(B)		-3.1248(B)	1.3439e+004(B)		
0.004	6.8305(R)	6.8356	-3.1416(R)	1.3491e+004(R)	-0.2034	0.2844
	6.8081(G)		-3.1639(G)	1.3560e+004(G)		
	6.8683(B)		-3.1038(B)	1.3374e+004(B)		
0.006	6.8259(R)	6.8420	-3.1462(R)	1.3505e+004(R)	-0.1925	0.2824
	6.8263(G)		-3.1458(G)	1.3504e+004(G)		
	6.8738(B)		-3.0982(B)	1.3357e+004(B)		
Speckle	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
0.002	6.7889(R)	6.7719	-3.1832(R)	1.3620e+004(R)	-0.1845	0.2956
	6.7599(G)		-3.2121(G)	1.3712e+004(G)		
	6.7671(B)		-3.2049(B)	1.3689e+004(B)		
0.004	6.8091(R)	6.7920	-3.1630(R)	1.3557e+004(R)	-0.1864	0.2962
	6.7752(G)		-3.1969(G)	1.3664e+004(G)		
	6.7917(B)		-3.1804(B)	1.3612e+004(B)		
0.006	6.8052(R)	6.7991	-3.1668(R)	1.3569e+004(R)	-0.1887	0.2957
	6.7873(G)		-3.1848(G)	1.3625e+004(G)		
	6.8048(B)		-3.1673(B)	1.3571e+004(B)		
Salt & Pepper	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
0.002	6.7046(R)	6.7042	-3.2674(R)	1.3887e+004(R)	-0.1847	0.3553
	6.7004(G)		-3.2717(G)	1.3901e+004(G)		

(contd...Table 1)

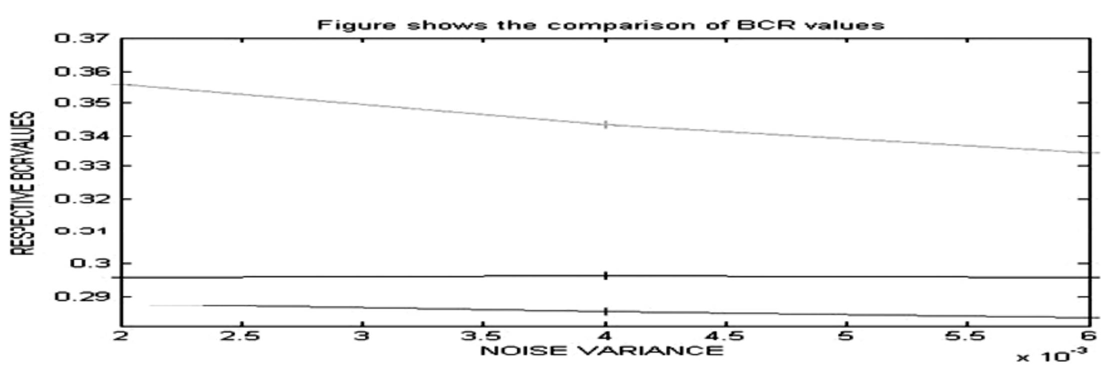
Gaussian	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
	6.7077(B)		-3.2644(B)	1.3878e+004(B)		
0.004	6.7144(R) 6.7093(G) 6.7210(B)	6.7149	-3.2576(R) -3.2628(G) -3.2511(B)	1.3856e+004(R) 1.3872e+004(G) 1.3835e+004(B)	-0.1848	0.3434
0.006	6.7189(R) 6.7094(G) 6.7333(B)	6.7205	-3.2531(R) -3.2627(G) -3.2388(B)	1.3842e+004(R) 1.3872e+004(G) 1.3796e+004(B)	-0.1860	0.3349



(a)



(b)



(c)

Figure 2: (a) PSNR Comparison (b) Corelation Coefficient comparison (c) BCR comparison between Watermark and Recovered Watermark

3.2.3. Using Red, Green and Blue components of Original image and Watermark image after applying 2 level DWT

Embedding Process

Step 1: Apply DWT to Original image

[LL1, HH1, HL1, HH1] = dwt2 (Original image);

Decomposed into 4 sub-bands: LL1, HH1, HL1 and HH1

Step 2: Apply DWT to LL1 sub-band

[LLL1, LLH1, LHL1, LHH1] = dwt2 (LL1);

Decomposed into 4 sub-bands: LLL1, LLH1, LHL1 and LHH1.

Step 3: Find Red, Green and Blue components of LLL1 sub-band of Original image.

Img1 = LLL1;

R1 = Img1 (:,:,1); G1 = Img1 (:,:,2); B1 = Img1 (:,:,3);

Step 4: Apply DWT to Watermark image.

[LLW1, LHW1, HLW1, HHW1] = dwt2 (Watermark image);

Decomposed into 4 sub-bands: LLW1, LHW1, HLW1 and HHW1.

Step 5: Apply DWT to LLW1

[LLLW1, LLHW1, LHLW1, LHHW1] = dwt2 (LLW1);

Decomposed into 4 sub-bands: LLLW1, LLHW1, LHLW1 and LHHW1.

Step 6: Find Red, Green and Blue components of LLW1 sub-band of Watermark image.

Img2 = LLLW1;

R2 = Img2 (:,:,1); G2 = Img2 (:,:,2); B2 = Img2 (:,:,3);

Step 7: Embed the Red, Green and Blue components of Img2 to Img1 with factor 0.080

R3 = R2+ (0.080*R1); G3 = G2+ (0.080*G1); B3 = B2+ (0.080*B1);

Step 8: Reconstruct LLL1 component by combining new and modified Red, Green and Blue components.

Step 9: Using Inverse DWT to combine [LLL1, LLH1, LHL1, LHH1] sub-bands and find LL1.

Step 10: Using Inverse DWT to combine [LL1, LH1, HL1, HH1] sub-bands and find Watermarked image.

Recovery Process

Step 1: Apply DWT to Watermarked image.

. [LLWED1, LHWED1, HLWED1, HHWED1] = dwt2 (Watermarked image);

Decomposed into 4 sub-bands: LLWED1, LHWED1, HLWED1 and HHWED1.

Step 2: Apply DWT to LLWED1.

[LLLWED1, LLHWED1, LHLWED1, LHHWED1] = dwt2(LLWED1);

Decomposed into 4 sub-bands: LLLWED1, LLHWED1, LHLWED1 and LHHWED1.

Step 3: Find Red, Green and Blue components of LLLWED1 sub-band of Original image.

Img3 = LLLWED1;

R4 = Img3 (:,:,1); G4 = Img3 (:,:,2); B4 = Img3 (:,:,3);

Step 4: Apply DWT to Original image.

[LL1, LH1, HL1, HH1] = dwt2(Original image);

Decomposed into 4 sub-bands: LL1, LH1, HL1 and HH1.

Step 5: Apply DWT to LL1.

[LLL1, LLH1, LHL1, LHH1] = dwt2(LL1);

Decomposed into 4 sub-bands: LLL1, LLH1, LHL1 and LHH1.

Step 6: Find Red, Green and Blue components of $LL1$ sub-band of Original image.

Img1 = LLL1;

R5 = Img1 (:,:,1); G5 = Img1 (:,:,2); B5 = Img1 (:,:,3);

Step 7: Reconstruct the Red, Green and Blue components of LLLW1 sub-band of Watermark image.

New Watermark Red= (Red component of LLWED1 - Red Component of LL1)/0.080;

New Watermark Green= (Green component of LLHWED1 - Green Component of LL1)/0.080;

New Watermark Blue= (Blue component of LHHWED1 - Blue Component of LL1)/0.080;

Step 8: Reconstruct the LLLW1 sub-band of Watermark image using modified Red, Green and Blue components.

Step 9: Combine the LLLW1, LLHW1, LHLW1, LHHW1 sub-bands using Inverse DWT and get LLW1 sub-band.

Step 10: Combine the LLW1, LHW1, HLW1 and HHW1 sub-bands using IDWT to get Recovered Watermark.

3.2.4. Results and Analysis

In this experiment, Red, Green and Blue components of LL sub-band of Original image and Watermark image are mixed after applying 2 level DWT. Watermark image has been recovered using recovery process steps of *3.2.3*. Fig 3(c) shows the Watermarked image and Recovered Watermark image has been shown in 3(d) respectively.

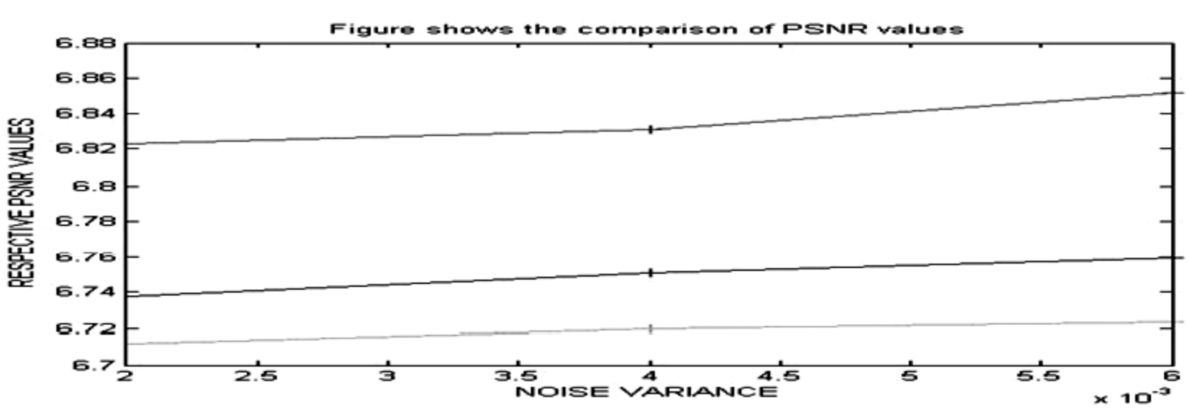
(a) **(b)** **(c)** **(d)**

Figure 3: (a) Original image (b) Watermark Image (c) Watermarked image (d) Recovered Watermark Image

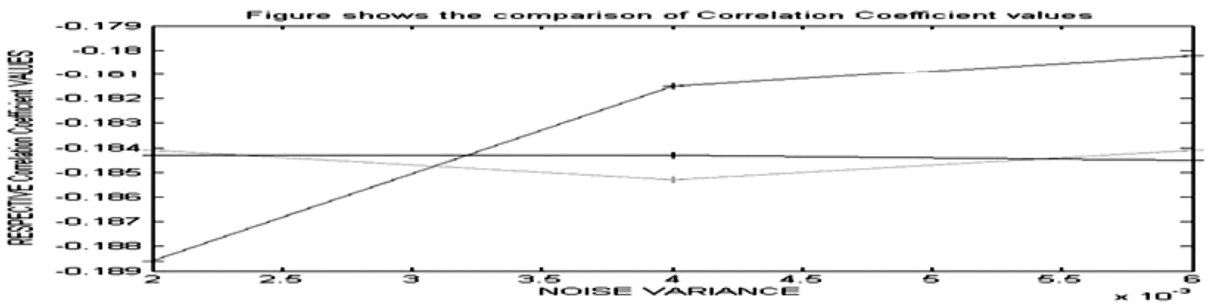
Table 2 shows the shows the PSNR, Avg. (PSNR), SNR, MSE, CC and BCR values against different quantity of Gaussian, Salt & Pepper and Speckle noise.

Table 2
Experiment Result

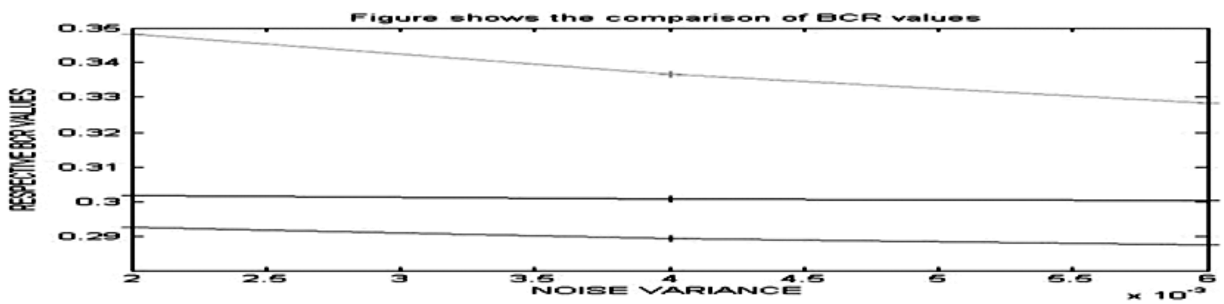
Gaussian	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
0.002	6.7918(R) 6.8366(G) 6.8402(B)	6.8228	-3.1802(R) -3.1354(G) -3.1318(B)	1.3611e+004(R) 1.3472e+004(G) 1.3460e+004(B)	-0.1886	0.2926
0.004	6.8123(R) 6.8374(G) 6.8428(B)	6.8308	-3.1598(R) -3.1347(G) -3.1292(B)	1.3547e+004(R) 1.3469e+004(G) 1.3452e+004(B)	-0.1815	0.2894
0.006	6.8291(R) 6.8562(G) 6.8686(B)	6.8513	-3.1430(R) -3.1159(G) -3.1035(B)	1.3495e+004(R) 1.3411e+004(G) 1.3373e+004(B)	-0.1802	0.2875
Salt & Pepper	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
0.002	6.7117(R) 6.7112(G) 6.7129(B)	6.7119	-3.2603(R) -3.2609(G) -3.2591(B)	1.3865e+004(R) 1.3866e+004(G) 1.3861e+004(B)	-0.1841	0.3483
0.004	6.7117(R) 6.7148(G) 6.7332(B)	6.7199	-3.2604(R) -3.2573(G) -3.2389(B)	1.3865e+004(R) 1.3855e+004(G) 1.3796e+004(B)	-0.1853	0.3366
0.006	6.7160(R) 6.7207(G) 6.7336(B)	6.7234	-3.2561(R) -3.2513(G) -3.2385(B)	1.3851e+004(R) 1.3836e+004(G) 1.3795e+004(B)	-0.1841	0.3283
Speckle	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
0.002	6.7491(R) 6.7303(G) 6.7342(B)	6.7378	-3.2229(R) -3.2417(G) -3.2378(B)	1.3746e+004(R) 1.3805e+004(G) 1.3793e+004(B)	-0.1843	0.3018
0.004	6.7632(R) 6.7420(G) 6.7480(B)	6.7510	-3.2088(R) -3.2301(G) -3.2241(B)	1.3701e+004(R) 1.3768e+004(G) 1.3749e+004(B)	-0.1843	0.3007
0.006	6.7626(R) 6.7545(G) 6.7603(B)	6.7591	-3.2094(R) -3.2175(G) -3.2118(B)	1.3703e+004(R) 1.3729e+004(G) 1.3710e+004(B)	-0.1845	0.3003



(a)



(b)



(c)

Figure 4: (a) PSNR comparison (b) Corelation Coefficient comparison (c) BCR comparison between Watermark and Recovered Watermark

3.2.5. Using 1 level DWT on Original image and Watermark image.

Embedding Process:

Step 1: Apply DWT to Original image.

[LL1, LH1, HL1, HH1] = dwt2 (Original image);

Decomposed into 4 sub-bands: LL1, LH1, HL1 and HH1.

Step 2: Apply DWT to Watermark image.

[LLW1, LHW1, HLW1, HHW1] = dwt2 (Watermark image);

Decomposed into 4 sub-bands: LLW1, LHW1, HLW1 and HHW1.

Step 3: Embed the LLW1 sub-band of the Watermark image into that of Original image with a factor of 0.10.

LL1_Modified = LL1 + 0.09*LLW1;

Step 4: By using the new LL1_Modified and [LH1, HL1, HH1] sub-bands of the Original image, recreate Watermarked image.

Recovery Process:

Step 1: Apply DWT to Watermarked image.

[LLWED1, LHWED1, HLWED1, HHWED1] = dwt2(Watermarked image);

Decomposed into 4 sub-bands: LLWED1, LHWED1, HLWED1 and HHWED1.

Step 2: Apply DWT to Original image.

[LL1, LH1, HL1, HH1] = dwt2 (Original image);

Decomposed into 4 sub-bands: LL1, LH1, HL1 and HH1.

Step 3: For reconstruction of Watermark, LL component of Recovered Watermark is created.

newwatermark_LL = (LLWED1 – LL1)/0.09;

Step 4: Recovered Watermark is recreated by using the newwatermark_LL and LHW1, HLW1, HHW sub-bands of the Watermarked image.

3.2.6. Results and Analysis

In another experiment 1 level DWT has been applied on Original image and Watermark image for watermarking. Figure 3(c) and 3(d) shows the Watermarked image and Recovered Watermarked image respectively after performing the experiment.To measure the performance of the above experiments some metrices such as Avg. PSNR, SNR, MSE, CC and BCR have been calculated on different noise with different variance.Table 3 shows the performance of the algorithm on gaussian, Salt & Pepper and Speckle noise with 0.002, 0.004 and 0.004.

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

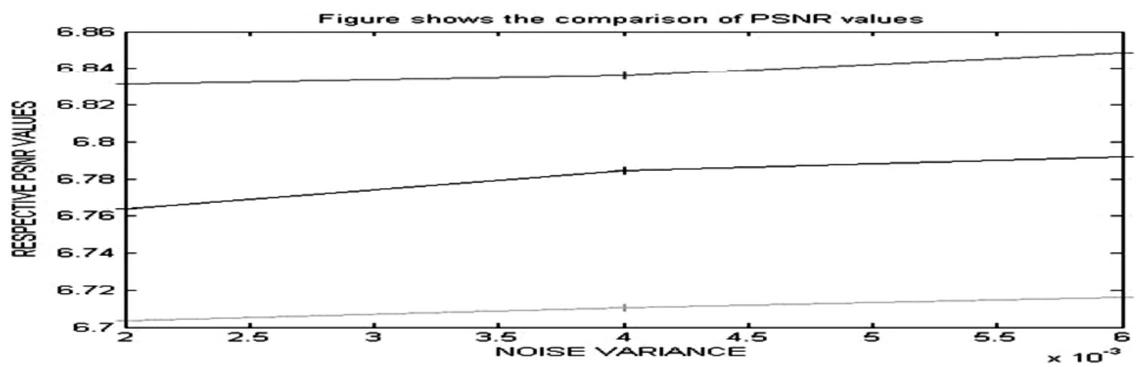
Figure 5: (a) Original image (b) Watermark Image (c) Watermarked image (d) Recovered watermark Image

Table 3
Experiment Result

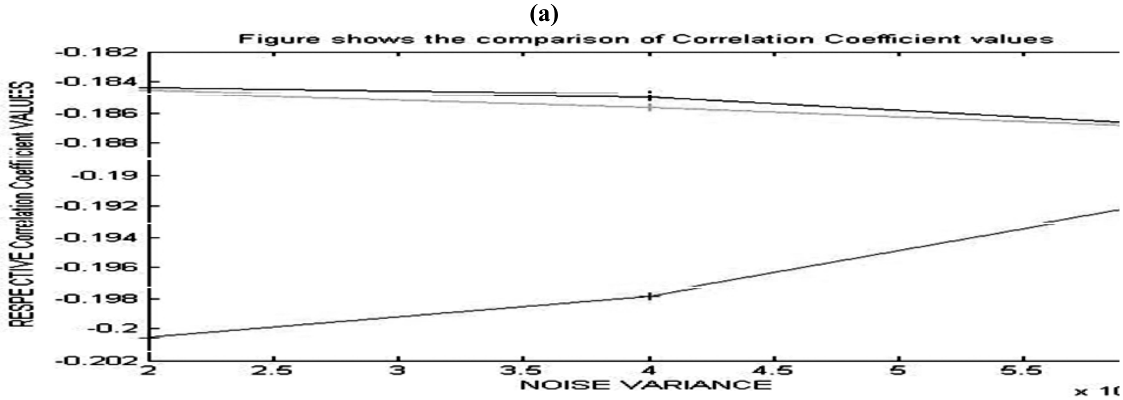
Gaussian	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
0.002	6.8143 (R) 6.8194(G) 6.8596(B)	6.8311	-3.1577(R) -3.1526(G) -3.1125(B)	1.3541e+004(R) 1.3525e+004(G) 1.3401e+004(B)	-0.2006	0.2868
0.004	6.8269(R) 6.8174(G) 6.8628(B)	6.8357	-3.1451(R) -3.1546(G) -3.1092(B)	1.3502e+004(R) 1.3531e+004(G) 1.3390e+004(B)	-0.1978	0.2855
0.006	6.8276(R) 6.8333(G) 6.8848(B)	6.8485	-3.1445(R) -3.1388(G) -3.0873(B)	1.3500e+004(R) 1.3482e+004(G) 1.3323e+004(B)	-0.1919	0.2834
Salt & Pepper	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*
0.002	6.7031(R) 6.6998(G) 6.7074(B)	6.7034	-3.2689(R) -3.2722(G) -3.2647(B)	1.3892e+004(R) 1.3903e+004(G) 1.3879e+004(B)	-0.1846	0.3528
0.004	6.7087(R) 6.7035(G) 6.7191(B)	6.7104	-3.2633(R) -3.2686(G) -3.2529(B)	1.3874e+004(R) 1.3891e+004(G) 1.3841e+004(B)	-0.1856	0.3376
0.006	6.7165(R) 6.7056(G) 6.7267(B)	6.7162	-3.2556(R) -3.2665(G) -3.2453(B)	1.3849e+004(R) 1.3884e+004(G) 1.3817e+004(B)	-0.1869	0.3364
Speckle	*PSNR*	*Avg. (PSNR)*	*SNR*	*MSE*	*CC*	*BCR*

(*contd...Table 3*)

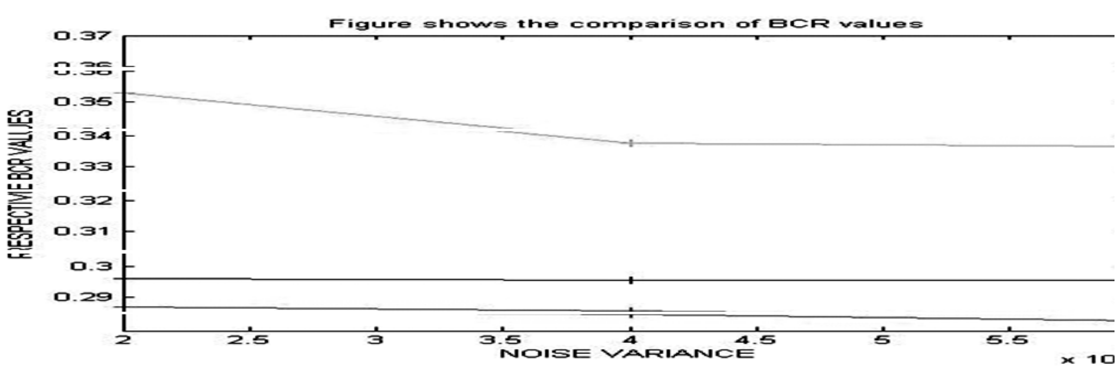
0.002	6.7816(R) 6.7507(G) 6.7593(B)	6.7638	-3.1905(R) -3.2214(G) -3.2128(B)	1.3643e+004(R) 1.3741e+004(G) 1.3714e+004(B)	-0.1844	0.2958
0.004	6.7981(R) 6.7720(G) 6.7828(B)	6.7843	-3.1739(R) -3.2000(G) -3.1892(B)	1.3591e+004(R) 1.3673e+004(G)	-0.1849	0.2953
0.006	6.7970(R) 6.7810(G) 6.7979(B)	6.7919	-3.1750(R) -3.1911(G) -3.1742(B)	1.3595e+004(R) 1.3645e+004(G) 1.3592e+004(B)	-0.1867	0.2953



(a)



(b)



(c)

Figure 6: (a) PSNR Comparison (b) Corellation Coefficient comparison (c) BCR comparison between Watermark and Recovered Watermark

4. CONCLUSION

This paper explains detail of DWT based image watermarking method and impact of noise with different variance on watermarked image. Here three DWT based watermarking algorithms are explained, one is color component and other is non-color component. The first algorithm is used first level of DWT and watermarking is done on color component of LL1 sub band. Second algorithm explained second level of DWT and watermarking is done on color component of LLL1. But in the third algorithm, first level of DWT and watermarking is done on LL sub- band. Second part of this paper is impact of noise on watermarked image, here author chose Gaussian, Salt & Pepper and Speckle noise with variance 0.002, 0.004, 0.006. For quality check, PSNR, SNR, MSE, BCR and correlation coefficient mathematical metrics have been used.

REFERENCES

[1] Tao, Hai, et al. “Robust image watermarking theories and techniques: A review.” Journal of applied research and technology 12.1 (2014): 122-138.

[2] Singh, Prabhishek, and R. S. Chadha. “A survey of digital watermarking techniques, applications and attacks.” International Journal of Engineering and Innovative Technology (IJEIT) 2.9 (2013): 165-175.

[3] G. N. Mohammed, A. Yasin and A. M. Zeki, “Digital Image Watermarking, Analysis of Current Methods,” 2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), Kuala Lumpur, 2012, pp. 324-329.

[4] J. Molina-García, R. Reyes-Reyes, V. Ponomaryov and C. Cruz-Ramos, “Watermarking algorithm for authentication and self-recovery of tampered images using DWT,” *2016 9th International Kharkiv Symposium on Physics and Engineering of Microwaves, Millimeter and Submillimeter Waves (MSMW)*, Kharkiv, pp. 1-4, 2016.

[5] S. Kamila, R. Roy and S. Changder, “A DWT based steganography scheme with image block partitioning,” *2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN)*, Noida, pp. 471-476, 2015.

[6] P. Bedi, V. Bhasin and T. Yadav, “2L-DWTS — Steganography technique based on second level DWT,” *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Jaipur, pp. 1533-1538, 2016.

[7] S. Kadu, C. Naveen, V. R. Satpute and A. G. Keskar, “Discrete wavelet transform based video watermarking technique,” *2016 International Conference on Microelectronics, Computing and Communications (MicroCom)*, Durgapur, pp. 1-6, 2016.

[8] R. Sharma, A. Kesarwani and P. M. Mathur, “A novel compression algorithm using DWT,” 2014 Annual IEEE India Conference (INDICON), Pune, 2014, pp. 1-4.

[9] C. T. Selvi, R. Sudhakar and G. Priyadharshini, “DWT based watermarking approach for medical image authentication,” 2016 10th International Conference on Intelligent Systems and Control (ISCO), Coimbatore, 2016, pp. 1-5.

[10] Ben Wang, Jinkou Ding, Qiaoyan Wen, Xin Liao and Cuixiang Liu, “An image watermarking algorithm based on DWT DCT and SVD,” 2009 IEEE International Conference on Network Infrastructure and Digital Content, Beijing, 2009, pp. 1034-1038.

[11] T. H. Nguyen, D. M. Duong and D. A. Duong, “Robust and high capacity watermarking for image based on DWT-SVD,” The 2015 IEEE RIVF International Conference on Computing & Communication Technologies - Research, Innovation, and Vision for Future (RIVF), Can Tho, 2015, pp. 83-88.

[12] Xiaoyi Zhou, Jixin Ma and Wencai Du, “SoW: A hybrid DWT-SVD based secured image watermarking,” PROCEEDINGS OF 2013 International Conference on Sensor Network Security Technology and Privacy Communication System, Nangang, 2013, pp. 197-200.

[13] Jane, Onur, E. Elbaþi, and H. G. Ýlk. “Hybrid non-blind watermarking based on DWT and SVD.” Journal of applied research and technology 12.4 (2014): 750-761.