International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol. 11, No. 1 (2018), pp.1-22
http//dx.doi.org/10.14257/ijsip.2018.11.1.01

Performance Evaluation of Spatial and Frequency Domain Techniques under Various Attacks

Bhavna Chauhan, Sangeeta Dhall and Shailender Gupta

YMCA, University of Science and Technology, Faridabad, India
bhavnachauhan2001@gmail.com, Sangeeta_dhall@yahoo.co.in, Shailender81@gmail.com

Abstract

Internet is very feasible and an excellent distribution system. One way to protect multimedia data against risk of illegal recording and malicious retransmission is to embed a signal, called watermark that authenticates the owner of the data. Watermark is an image or text to be embedded into the documents that needs protection. A good watermarking scheme is the one which can offer resistance against attacks and is imperceptible to hackers. Robustness measures the withstanding capability of watermark against various attacks. This paper is an effort to study different watermarking techniques on the basis of given parameters. For the purpose of implementation, MATLAB is used and comparison is done on the basis of various performance metrics such as PSNR, MSE, Time Complexity, Correlation Co-efficient and robustness against various attacks like cropping, Gaussian noise, salt and pepper noise etc. The result shows that frequency domain techniques are comparatively more robust with DWT being the most robust one.

***Keywords**: Correlation Coefficient, DWT, Gaussian noise, MSE, PSNR, Robustness, Salt and pepper noise, Watermarking*

1. Introduction

In present scenario, Digitalization has become inseparable part of our lives and every possible move is being taken in this direction across the globe. It has resulted in a more handy system for online transaction and data sharing over Internet. Moreover, Internet has become the quickest and the most user friendly way of transferring data. But this hike in its usage has adversely resulted in the threats of piracy and copyright of the information content. This growing concern over secure transmission led to the advent of Watermarking techniques which are superior to cryptography [1] and steganography [2, 3,4] techniques in terms of robustness.

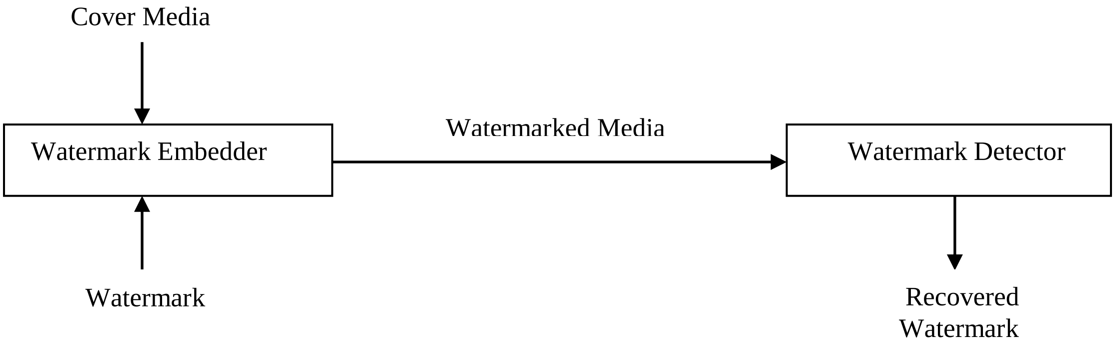


Figure 1. Watermarking Mechanism

Received (May 21, 2017), Review Result (November 24, 2017), Accepted (December 1, 2017)

ISSN: 2005-4254 IJSIP

Digital watermarking [5-8] is the process of hiding or embedding the important data such as audio, video or image into a carrier signal (see Figure. 1.).

Digital Watermarking is used for various purposes ranging from authentication, broadcast monitoring, owner identification, content protection and copyright protection, tamper detection, medical applications *etc* [9-11]. It is categorized on the basis of robustness as Robust, Fragile and Semi-fragile watermarking. A watermark is called robust if it resists a specific set of transformations. On the other hand, if watermark fails to be detected after slightest of modification, it is known as Fragile. A Semi-fragile watermark is the one which can withstand mild transformations but not the malignant ones (see Figure. 2.) [12-14].

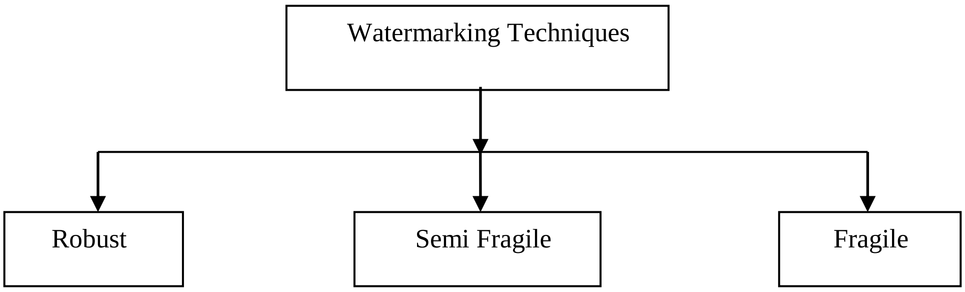


Figure 2. Classification of Watermarking Techniques

Watermarking techniques are performed in two domains- Spatial and Frequency. In spatial domain, watermark is embedded by modifying some selected pixels, hence changing the intensity and the colour value. The algorithm should carefully weigh the number of changed bits in the pixels against the possibility of the watermark becoming visible. The watermark can be detected by correlating the expected pattern with the received signal. These primitive techniques are vulnerable to attacks, noise and common signal processing and the watermark can be easily destroyed [15]. In the Frequency domain watermarking techniques, firstly the original image is converted by a predefined transformation like DCT, DWT *etc*. Then the watermark is embedded in the transformation coefficients. Finally, the inverse transform is performed to obtain the watermarked image. These techniques have an advantage over spatial domain techniques as they hide information in areas of the image that are less exposed to compression, cropping, and image processing [16].

This paper is a survey of various techniques on the basis of Robustness. The rest of the paper is organized as follows: Section II contains the watermarking techniques used; Section III contains the performance metrics; Section IV contains the results; Section V contains the overall conclusion followed by the references.

2. Watermarking Techniques

As mentioned earlier, a robust watermarking scheme is the one in which watermark remains unaffected, in spite of transformations and malicious attacks. Typical image degradations are rotation, cropping, additive noise, JPEG compression and quantization. Such watermarks find use in copy protection applications to carry copy and no access control information. There are different techniques of embedding the message in the carrier signal like LSB substitution, pseudo random LSB, Distortion, DCT, DWT, DWT-SVD *etc*. These techniques are discussed below in details:

2.1. Spatial Domain Techniques

2.1.1. LSB Watermark Substitution

In LSB substitution, the cover image is converted into binary form. Similarly, pixels of watermark data are also converted in the form of a stream of bits. Then, the least significant bit of pixels of the cover image is substituted with the bits of watermark data. Hence, the pixels are modified and a new set of pixels is obtained to form watermarked image. This results in an unrecognizable change in the image which is why this is known as imperceptible watermarking scheme. On the Receiver side, the watermark can be retrieved by extracting the LSBs from the watermarked image and rearranging them in the same fashion as pixel values [17].

Embedding algorithm:

Sender side-

```
a=imread (X);                    /*Read the cover image (X)
b=imread(W);                     /*Read the watermark image (W)
b1=rgb2gray(b);                  /*Convert RGB watermark to grayscale
ls=bitget(a,1);                  /*Extract LSB of cover image (X)
b2=de2bi(b1,8);                  /*Convert decimal to binary
b3=reshape(b2,1,len*bre);        /*Reshape the watermark (W)
e=1;
for k=1:p                        /*To access each pixel of cover image (X)
for i=1:m
for j=1:n
if(e<=len*bre)  /*Up to the size of watermark (W)
        ls(i,j,k)=b3(e);  /*Substitute the watermark bit in LSB of pixels
    e=e+1;
                  else
                  break;          /*Stop the loop
 end          /*End if
     end                 /*End for
     end                         /*End for
end                              /*End for
a1=bitset(a,1,ls);               /*Replace the modified LSB (ls) back in the cover image
```

Receiver side-

```
l1=bitget(a1,1);                 /*Extract  LSB from watermarked image
f=1;
for k=1:p                        /*Extract each pixel
for i=1:m
for j=1:n
if(f<=len*bre)    /*Up to the size of watermark
 z(f)=l1(i,j,k);     /*Combine the LSBs to form watermark
                 f=f+1;
else
                 break;          /*Stop the loop
end
end             /*End for
end                       /*End for
end                              /*End for
y=reshape(z,u*v,8);              /*Reshape it to  form pixels
h=bi2de(y);                      /*Convert binary to decimal
g=reshape(h,u,v);                /*Reshape back to original dimension
```

LSB substitution method is the simplest approach and has the advantage of good picture quality, high embedding capacity with ease of implementation. But it is very less robust and hence prone to manipulations.

2.1.2. Pseudorandom LSB Watermark Substitution

Pseudo Random LSB substitution technique is one of the most widely used techniques. In this technique, a random-key is generated by pseudo random generator that is used to choose the pixels randomly where watermark data can be embedded. This will make it more difficult for an intruder to find the watermark data. Moreover, for the colored image we have three planes (RGB) and random number generator can choose pixel from any of the planes. Due to this, it will be further more difficult for the attacker to identify the pattern in which watermark is hidden owing to the fact that no particular pattern is followed for embedding. At transmitter side a random key is used to randomize the cover image and embed the watermark bits into the LSB of the pixels. The transmitting and receiving end share the random-key. This random-key is used as a seed for pseudo-random number generator for selecting pixel locations in an image for hiding the watermark [18].

Embedding algorithm:

Sender side-

```
a=imread (X);                    /*Read the cover image (X)
b=imread(W);                     /*Read the watermark image (W)
b1=rgb2gray(b);                  /*Convert RGB watermark to grayscale
a1=a(:,:,1);                     /*Extract the pixels of red plane of cover image(X)
a2=reshape(a1,1,m*n);            /*Reshape it  into one dimensional array
ls=bitget(a2,1);                 /*Extract LSB of cover image (X)
b2=de2bi(b1,8);                  /*Convert decimal to binary
b3=reshape(b2,1,len*bre);        /*Reshape the watermark (W)
A=random(a,b);                   /*Randomize pixels using seed
j=1;
while(j<=m*n)                    /*Up to the size of cover image
for i=1:u*v*8         /*For each bit of pixels of watermark
if(A(i)==j)
a2(j)=bitset(a2(A(i)),1,b3(i));/* Substitute the watermark bit
break;                /*Stop the loop
end                   /*End if
end                   /*End for
j=j+1;
end                              /*End for
d=reshape(a2,m,n);               /*Reshape the obtained image to original dimension
a(:,:,1)=d;                      /*Substitute the modified plane back to the cover image
```

Receiver side-

```
a3=a(:,:,1);                          /*Extract the pixels of red plane of watermarked
image(Y)
a4=bitget(a3,1);                 /*Extract  LSB from watermarked image
a5=reshape(a4,1,m*n);            /*Reshape it  into one dimensional array
B=random(a,b);                   /*Randomize pixels using seed
for i=1:u*v*8                    /*For each bit of pixels of watermark
z(i)=bitget(a5(B(i)),1);         /*Extract LSB  from corresponding pixel
end                              /*End for
y=reshape(z,u*v,8);              /*Reshape to form pixels
h=bi2de(y);                      /*Convert binary to decimal
g=reshape(h,u,v);                /* Reshape back to original dimension
```

It too retains good picture quality after embedding and also has high embedding capacity with ease of implementation but it is less robust. Moreover, if the intruders have access to seed the data can be manipulated easily.

2.2.3. Distortion based Watermarking

In distortion technique, value of pixel is changed for hiding the watermark media. This change in value is known as distortion. Pixel values are randomly chosen (pseudo-randomly) for inserting watermark bits in cover image. Algorithm under this technique provides deviation in the pixel value for hiding this watermark information. To embed bit 1, deviation value x is added or subtracted from pixel's value. On reception, watermark retrieval from cover image is done by using the same deviation. Under this algorithm, the original cover image is the fundamental prerequisite at the receiver side for recovery of watermark information. This information can be retrieved by calculating deviation between cover and watermark image [19, 20].

Embedding algorithm:

Sender side-

```
function transmission (cover_img, secret_img)
secret_img_bin = dec2bin(secret_img);
d= cover_img;
while size_secret_msg is reached
for all cover_img(random_sequence)
if secret_msg (loop)==1
if cover_img(pixel)<=128;
d(pixel)= cover_img (pixel)+1;
else
d(pixel)= cover_img (pixel)-1;
end
end if
end for
end while
stego_img = d;
```

Receiver side-

```
input : stego image (stego_img), cover image (cover_img), random sequence (random_sequence);
output : secret message in binary form (secret_msg),
function retrieval (stego_img, cover_img, random_sequence)
for all stego_im(random_sequence)
difference = stego_im(random_sequence)- cover_im(random_sequence)
if difference == 0
secret_msg=0;
else if difference == 1
secret_msg=1;
end if
end for
```

The image quality is preserved in the watermarked image and it is also secure because of the random distribution of the watermark bits in cover image. The disadvantage associated with it is lack of robustness.

2.2. Frequency Domain Techniques

2.2.1. DCT Watermarking Substitution

In Discrete Cosine Transform technique, cover image or signal is converted into frequency domain from spatial domain. This transformation process include following phases.

1. Initially the image or the cover signal is divided into 8x8 pixel size blocks.
2. DCT transformation is applied to each block to convert the information into frequency domain.
3. Now data is embedded in the frequency components after which compression of the signal takes place by removing the unwanted part [21, 22].

Embedding algorithm:

Sender side-

```
function stegano(cover_img, secret_img)           /*Read cover image
secret_img_bin = dec2bin(secret_img);             /*Read secret message and convert it into binary
cover_img_blocks = img2block(cover_img,8,8);   /*Divide the cover image into 8X8 blocks
cover_img_blocks = cover_img_blocks-128;     /*Working from the left to right, top to bottom, subtract 128
cover_img_blocks_dct = dct2(cover_img_blocks); /*Apply DCT to each block
cover_img_blocks_dct_quantised = quant(cover_img_blocks_dct);     /*Compress each block using compression table
stego_img = lsb_substitution(cover_img_blocks_dct_quantised,secret_img_bin);    /*Calculate LSB of each DC coefficient and replace with each bit of secret message.
```

The same process can be carried out in reverse order to retrieve the watermark on Receiver side. As the watermark data is embedded in the transformed image, DCT watermarking is highly robust against most of the attacks, but it suffers the drawbacks of low embedding capacity and also poor picture quality of watermarked image.

2.2.2. DWT Watermarking Substitution

Discrete Wavelet Transform (DWT) technique is another frequency domain transformation proposed by Haar. In Haar-DWT, initially, scanning of the pixels from left to right is done in the horizontal direction *i.e.* row wise. During this horizontal scanning of the pixels, the addition and subtraction on the adjacent pixel values are performed. Then fine details in small area are recorded in which pixel addition denotes the lower frequency component (L) and subtraction represents higher frequency components (H). In the similar fashion, scanning of the pixels is done from top to bottom in vertical direction *i.e.* column-wise and again addition and subtraction operations are performed for adjacent pixels. After the operation in the vertical direction, the sum is stored in the upper half and the difference in the lower half and matrix is obtained [7, 23, 24].

Embedding algorithm:

Sender side-

```
a1=imread(X);                      /*Read the cover image (X)
a=rgb2gray(a1);                    /*Convert RGB cover to grayscale
a=a/255;                           /*Normalize the cover image
[cA,cH,cV,cD]=dwt2(a,'haar');      /*Apply Haar 2-D DWT transformation on cover image
b=imread(W);                       /*Read the watermark image (W)
b=rgb2gray(b);                     /*Convert RGB watermark to grayscale
b1=de2bi(b,8);                          /*Convert the pixels of watermark from decimal to binary
b2=reshape(b1,1,len*bre);          /*Reshape it into one dimensional array
i1 = randi(50,1,mlen);             /*Generate a PN sequence
k=1;
 j=1;
for i=1:1:mlen                     /*Run Up to to the entire length of sequence
if (m(i)==0)
```

```
        cH1(k,j)=cH(k,j)+i1(i);      /*Add PN sequence to H1 components when  message bit is 0
        end                          /*End if
 j=j+1;
if (j>n)
k=k+1;
 j=1;
 end                      /*End for
 end                                 /*End for
Y=idwt2(cA,cH1,cV,cD,'haar');        /*Take inverse Haar 2D-DWT
Y=Y*255;                             /*Reverse the normalization
```

Receiver side-

```
Y1=Y/255;                            /*Normalize the watermarked image
[cA2,cH2,cV2,cD2] = dwt2(X1,'haar'); /* Apply Haar 2-D DWT transformation
for i=1:1:mlen                       /*Run Up to the entire length of PN sequence
msg(i)=1;                 /*Initialize message to all ones
end                                  /*End for
k=1; j=1;
for i=1:1:mlen
if (cH2(k,j)~=cH(k,j))    /*Comparing the two values
        msg(i)=0;                    /*message bit is interpreted as 0 in this case
end                       /*End if
j=j+1;
if (j>n)
k=k+1;
 j=1;
end                       /*End for
 end                                /*End for
y=reshape(msg,len,8);               /*Reshape the message vector to form pixels
h=bi2de(y);                         /*Convert binary to decimal
h1=uint8(h);                        /*Return the integer value
g=reshape(h1,u,v);                  /*Reshape back to original dimension
```

DWT method gives better image quality as compared to the DCT. It is highly robust as it has endurance against wavelet transform based image compression as well as to other common image distortions like rescaling, half toning, additive noise *etc*. But it is more complex to implement and hence time consuming.

2.3. DWT-SVD based Watermarking

This technique is a combination of discrete wavelet transform (discussed in section 2.2.2) and Singular value decomposition (SVD) in YCbCr color space. SVD is a linear algebra transform with a number of applications of image processing. SVD of an image M with dimensions m x m is given by:

$$M = USV^T$$

where, U and V are orthogonal matrices and S is a diagonal matrix carrying non-negative singular values of matrix M. The columns of U and V are called left and right singular vectors of M respectively. Here, U represents the horizontal details while V represents the vertical details of the original image. The diagonal values of Matrix S are arranged in decreasing order which depicts the decreasing significance of entries from first to last. This feature is used in SVD based compression methods [25]. There are two main properties of SVD employed in digital watermarking scheme.

- Small variations in singular values do not affect the quality of image.

- Singular values of an image have high stability which ultimately leads to high tolerance against various attacks.

Embedding algorithm:

Sender side-

```
a=imread(X);                          /*Read the cover image(X)
a1=rgb2ycbcr(a);                      /*Convert RGB image to YCbCr color space
a2=a1(:,:,1);                         /*Extract Y color plane
[cA1,cH1,cV1,cD1]=dwt2(a2,'haar');    /*Perform  4 level 2D-DWT to subdivide the image
[cA2,cH2,cV2,cD2]=dwt2(cD1,'haar');
[cA3,cH3,cV3,cD3]=dwt2(cD2,'haar');
[cA4,cH4,cV4,cD4]=dwt2(cD3,'haar');
[cU,cS,cV]=svd(cD4);                  /*Select cD4 channel and apply SVD to it
b=imread(W);                          /*Read watermark image(W)
b1=b(:,:,1);                          /*Extract R color plane
b2=arnoldtrans(b1);                   /*Apply Arnold transform to scramble R color plane
[wcA1,wcH1,wcV1,wcD1]=dwt2(b2,'haar');     /* Perform  3 level 2D-DWT
[wcA2,wcH2,wcV2,wcD2]=dwt2(wcD1,'haar');
[wcA3,wcH3,wcV3,wcD3]=dwt2(wcD2,'haar');
[wU,wS,wV]=svd(wcD3);                 /* Select wcD3 channel and apply SVD to it
Snew=cS.*0.9+wS1.*0.3;                /*Embed the watermark using alpha blending equation
s=cU*Snew*cV;                         /*To obtain new s
m1=idwt2(cA4,cH4,cV4,s,'haar');     /*Apply inverse 4 level DWT to s to obtain Y channel watermarked image
m2=idwt2(cA3,cH3,cV3,m1,'haar');
m3=idwt2(cA2,cH2,cV2,m2,'haar');
m4=idwt2(cA1,cH1,cV1,m3,'haar');
a1(:,:,1)=m4;                         /* Substitute the new value of color channel
a6=ycbcr2rgb(a1);                     /*Convert RGB to YCbCr color space
```

Receiver side-

```
a=imread(X);                          /*Read the cover image(X)
c1=rgb2ycbcr(a);                      /*Convert RGB image to YCbCr color space
c2=c1(:,:,1);                         /*Extract Y color plane
[dA1,dH1,dV1,dD1]=dwt2(c2,'haar'); /*Perform 4 level 2D-DWT to subdivide the image
[dA2,dH2,dV2,dD2]=dwt2(dD1,'haar');
[dA3,dH3,dV3,dD3]=dwt2(dD2,'haar');
[dA4,dH4,dV4,dD4]=dwt2(dD3,'haar');
a4=rgb2ycbcr(a3);                      /*Convert  RGB image to YCbCr color space
a5=a4(:,:,1);                          /* Extract Y color plane
[wmA1,wmH1,wmV1,wmD1]=dwt2(a5,'haar');    /*Perform 4 level 2D-DWT
[wmA2,wmH2,wmV2,wmD2]=dwt2(wmD1,'haar');
[wmA3,wmH3,wmV3,wmD3]=dwt2(wmD2,'haar');
[wmA4,wmH4,wmV4,wmD4]=dwt2(wmD3,'haar');
d=imread(W);                          /*Read the watermark image
d1=d(:,:,1);                          /*Extract R color plane
d2=arnoldtrans(d1);                   /*Apply Arnold transform to scramble R color plane
[WcA1,WcH1,WcV1,WcD1]=dwt2(d2,'haar');      /*Perform 3 level 2D-DWT
  [WcA2,WcH2,WcV2,WcD2]=dwt2(WcD1,'haar');
  [WcA3,WcH3,WcV3,WcD3]=dwt2(WcD2,'haar');
[WU,WS,WV]=svd(WcD3);                 /*Select  WcH3 and apply SVD to it
[dU,dS,dV]=svd(dD4);                  /*Select dD4 and apply SVD to it
[wmU,wmS,wmV]=svd(wmD4);              /*Select wmD4 and apply SVD to it
SNEW=(wmS-0.3*dS)/0.9;                /*Extract watermark using alpha blending equation
WV1=transpose(WV);                    /*Take transpose of WV
 S=WU1*SNEW*WV1;                      /*Obtain new S using SNEW
 M1=idwt2(WcA3,WcH3,WcV3,S,'haar');/*Perform  inverse 3 level DWT  on S
 M2=idwt2(WcA2,WcH2,WcV2,M1,'haar');
```

```
M3=idwt2(WcA1,WcH1,WcV1,M2,'haar');
x=antiarnold(M3);                    /*Apply anti Arnold transform to unscramble the image
d(:,:,1)=x;                          /*Substitute the new value of color channel
a6=ycbcr2rgb(d);                     /*Convert YCbCr to RGB color space
```

YCbCr color space has been used in this method which has following advantages-

- Embedding watermark in this space gives more robust and imperceptible watermarked image.
- RGB Color Channel is complex in describing the color pattern and has redundant information between each component.

Scrambling in pre-processing and post-processing add to the security of this method. The only drawback is its complexity.

3. Experimental Setup

3.1. Setup Parameters

Table 1. Simulation Setup Parameters

Processor	Intel® Core(TM)i3-3217UCPU
Memory	4.00 GB
Operating System	Windows 8 Pro (64 bit)
Tool used	MATLAB
Version	7.10.0
Images Type	Jpg
Resolutions of Cover Images	Min: 64X64 Max: 1024x1024
Cover Images Size	Min: 3.22 KB Max: 17.3 KB
Resolution of Watermark	14X20
Watermark size	1.41B
Cropped Area	10X10, 20X20, 30X30
Noise Density Variation	0-1

3.2. Performance Analysis test and Parameters

3.2.1. Mean Square Error (MSE)

The MSE is obtained as a cumulative of the square of the errors between the image obtained after watermarking and the original image. Lower the value of MSE means lower is the error.

$$\mathrm{MSE} = \frac{1}{\mathrm{size}} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [f(i,j) - K(i,j)]^2$$

where,

f(i,j) is the pixel value of original image,

K(i,j) is the pixel value of newer image(noisy approximation), size of image is m x n for monochrome image and mxnx3 for colored image.

3.2.2. Peak Signal Noise Ratio (PSNR)

It is defined as the ratio between the maximum possible power of a signal and the power of corrupting noise. Or, it is the ratio of peak square value of pixels by mean square error (MSE). It is expressed in decibel (db). PSNR is a good measure for comparing restoration results for the same image. The PSNR is defined as:

$$\mathrm{PSNR} = 10.\log_{10}\left(\frac{\mathrm{MAX_I}^2}{\mathrm{MSE}}\right)$$

where, $\mathrm{MAX_I}$ represents maximum value of pixel of the image,
MSE is the mean square error.

3.2.3. Correlation Coefficient

This parameter is a measure of the linear correlation i.e. dependence between two images A and B. Its range lies between -1 to +1 both inclusive, where 1 signifies perfect match and −1 signifies total mismatch. Correlation Coefficient (CC) parameter identifies the association among two images. The CC between original image and watermarked image computes image deformation at pixels level. The correlation coefficient can be calculated as:

$$\rho(A, B) = \frac{\mathrm{cov}(A, B)}{\sigma_A \sigma_B}$$

where, A is cover image and B is the watermarked-image, (A, B) is the correlation coefficient between image matrices A and B.
Cov(A,B) is the covariance between matrices A and B,
σ_A is the standard deviation of A, σ_B is the standard deviation of B.

3.2.4. Bit Error Rate

Bit error rate is defined as number of erroneous bits transmitted per unit time. BER is a key parameter that is used in assessing systems that transmit digital data from one location to another. During transmission of digital data over a communication channel, there may cause alteration of bits occurs due to noise, interference, *etc*. Thus, there is need to calculate BER. This BER increases with the decrease in quality of channel.

$$BER = \frac{Number\ of\ bit\ errors}{Number\ of\ transmitted\ bits}$$

3.2.5. Cropping Attack

Cropping refers to the process of removing the outer portions of an image for improving framing, highlighting subject matter or changing aspect ratio. Effect of cropping attack can be seen on watermark image for different values of aspect ratio.

3.2.6. Salt and Pepper Noise

It is one of the different forms of noise. It is an external disturbance sometimes seen on the images. It is also known as Impulse Noise as it is caused by sharp and sudden disturbances in the image signal. Its presence can be observed as occurrence of white and black pixels on the image. Effect of Salt and Pepper noise can be seen on watermark image for varying values of noise density.

3.2.7. Gaussian Noise

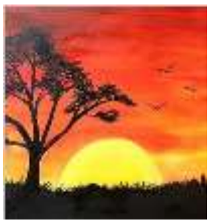
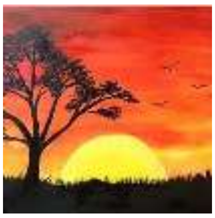
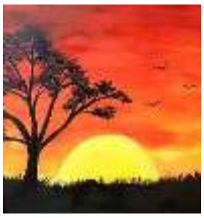
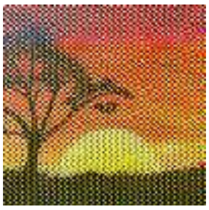
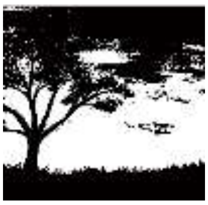
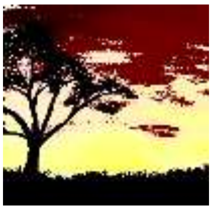
It is very prominent noise caused by the random fluctuations in the signal. It is a statistical noise which is defined as normal or Gaussian distribution *i.e.* probability density function (PDF) p can be defined as:

$$p(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(z-\mu)^2}{2\sigma^2}}$$

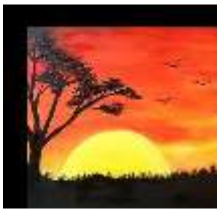
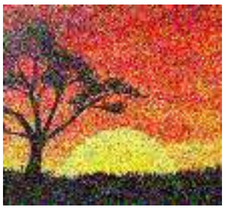
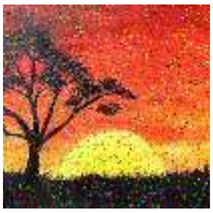
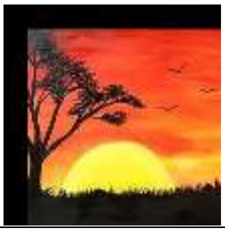
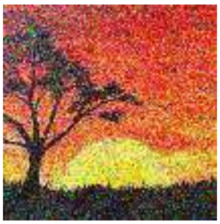
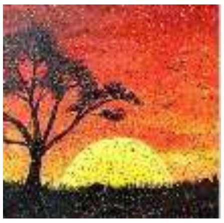
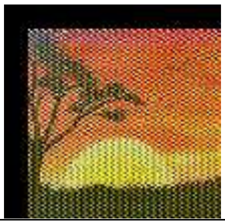
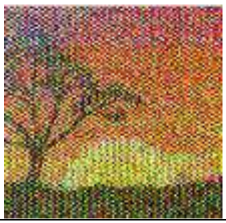
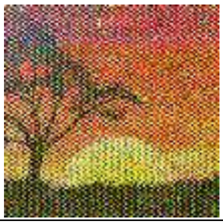
where, z represents the grey level, μ the mean value and σ the standard deviation. Effect of Gaussian noise can be seen on watermark image for different values of mean and standard deviation.

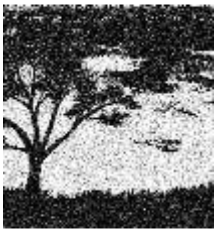
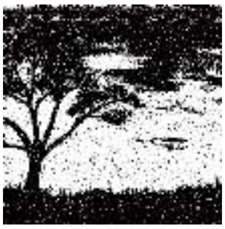
3.3. Snapshots

3.3.1. Watermarked Image under Different Techniques

Cover Image (256X256)	Spatial domain techniques		
	LSB	Pseudorandom	Distortion
			
Watermark Image (14X20)	Frequency domain techniques		
	DCT	DWT	DCT-SVD
			

3.3.2. Watermarked Image under Various Attacks

	Cropping (10X10)	Gaussian Noise (0.1)	Salt & Pepper Noise(0.1)
LSB Substitution			
Pseudorandom Substitution			
DCT			

DWT			

4. Results

4.1. Impact on MSE

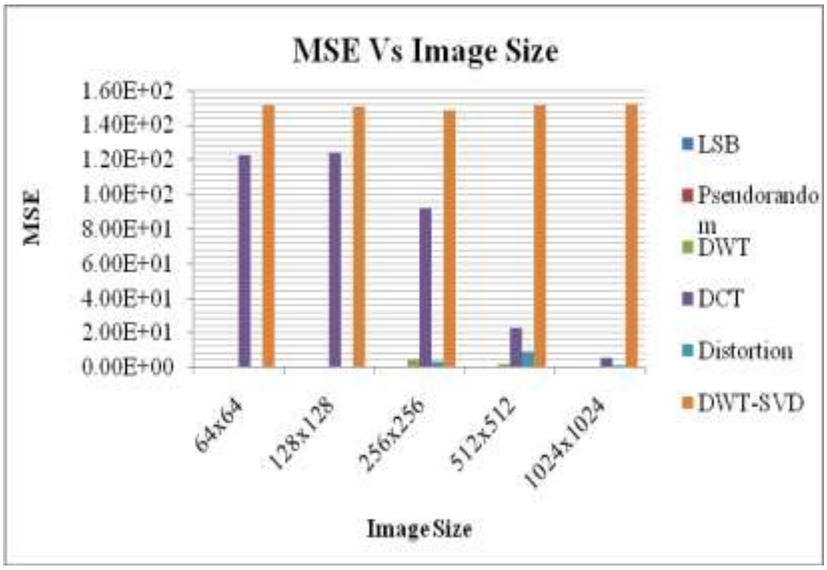


Figure 3. MSE Vs Image Size

- MSE is very high for DCT and DWT-SVD because they are frequency transformation techniques, so the watermark is embed into cover image after being transformed into frequency domain.
- Though DWT also involves transformation but it shows moderate values of error.
- In spatial domain techniques, only pixel values of cover image are changed with watermark. Thus, MSE remains quite low for LSB and Pseudorandom (see Figure 3.).

4.2. Impact of Cropping on MSE

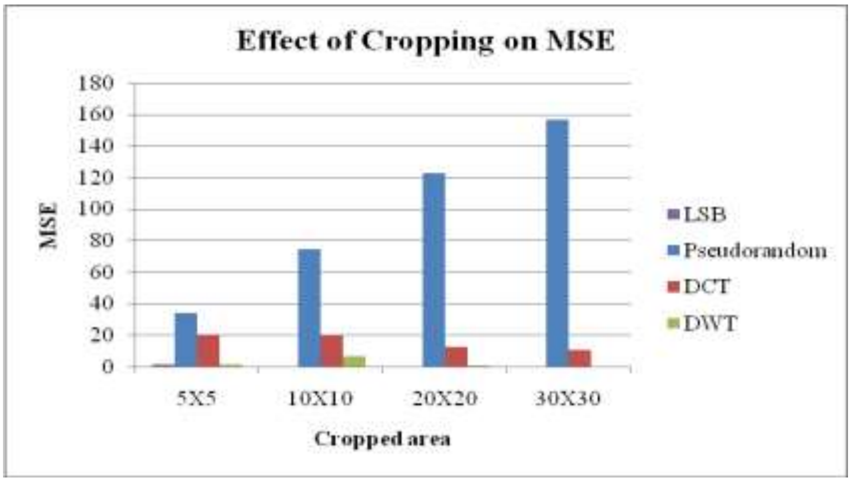


Figure 4. MSE Vs Cropped Area

- The effect of cropping is most prominent in the Mean Square error of Pseudorandom, which shows a drastic increase. It is due to the random distribution of watermark bits.
- DCT shows moderate values.
- Other techniques are not much affected, especially DWT where the MSE is least after cropping (see Figure 4.).

4.3. Impact of Gaussian Noise on MSE

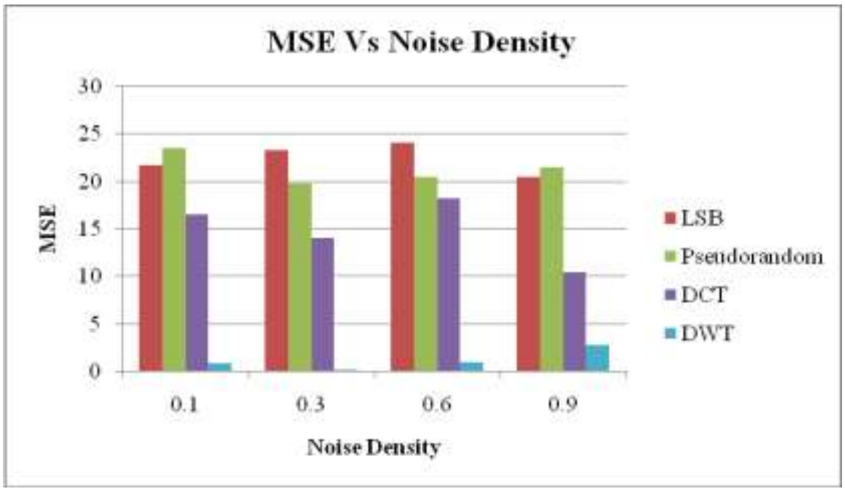


Figure 5. MSE Vs Gaussian Noise Density

- Gaussian noise is found to have much impact on mean square error in spatial domain techniques where it shows appreciably high values at all densities.
- DCT shows moderate values.
- MSE is recorded least in DWT, where it shows minimal error in retrieved watermark (Refer Figure 5.).

4.4. Impact of Salt and Pepper Noise on MSE

*****Cropping attack on LSB substituted watermark leads to the removal of area where the watermark bits are embedded in cover image, which in turn causes null values in last two observations.

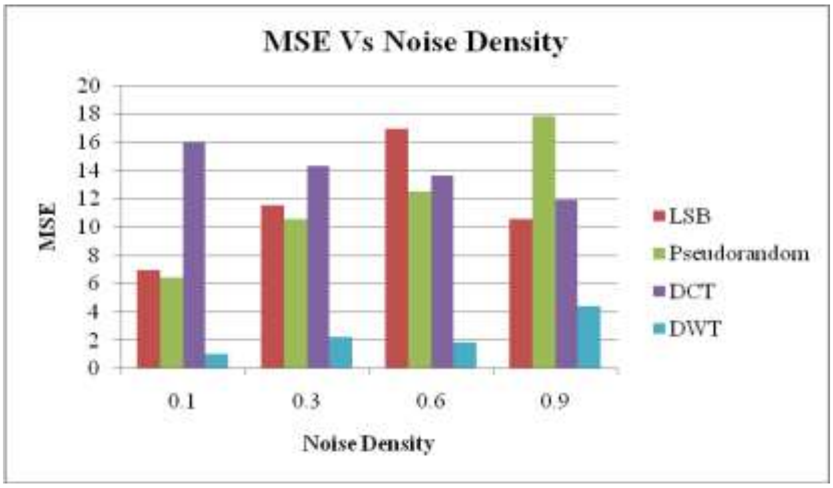


Figure 6. MSE Vs Salt and Pepper Noise Density

- Referring to Figure 6, it can be seen that MSE is highest in DCT at lower noise densities while spatial domain techniques slowly take over as the density increases.
- It can be observed that application of Salt and pepper noise affects the retrieval of watermark in DWT the least as MSE has minimum values in this technique.

4.5. Impact on PSNR

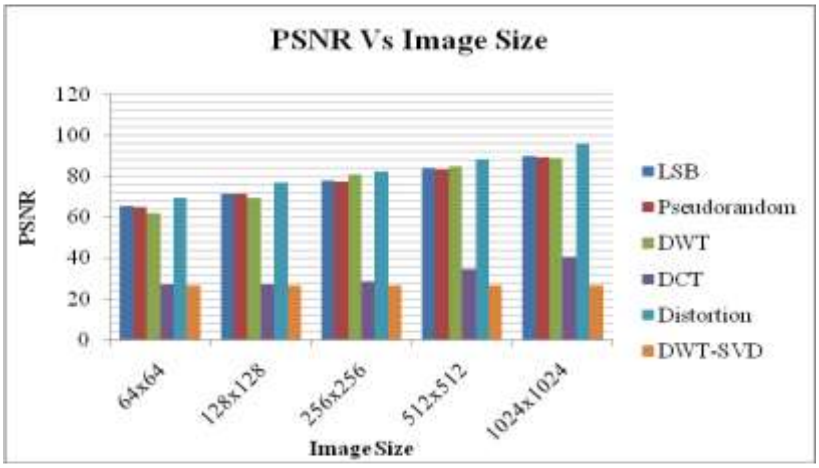


Figure 7. PSNR Vs Image Size

- It can be observed that PSNR is highest for Distortion followed by DWT.
- LSB and Pseudorandom also have comparable values of PSNR owing to low MSE in these methods. It is lowest for DWT-SVD as the image is deteriorated due to frequency transformation at large scale (see Figure 7.).
- This result is also confirmed by the maximum value of MSE seen in DWT-SVD. As PSNR is inversely proportional to MSE, this result is justified.

4.6. Impact of Cropping on PSNR

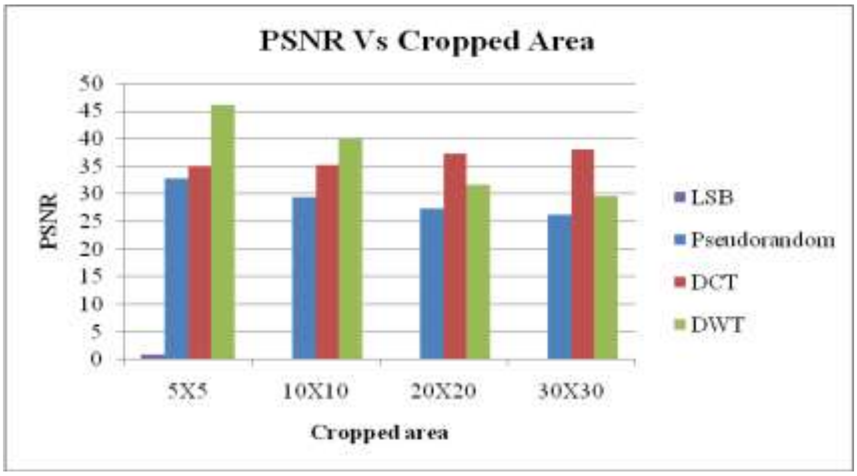


Figure 8. PSNR Vs Cropped Area

- PSNR of DWT comes out to be highest for smaller cropped area while as we increase it, DCT shows largest PSNR value (Refer Figure 8.).
- Pseudorandom shows moderate values of PSNR.
- LSB results into negligible values because of the loss of embedded part due to cropping.

4.7. Impact of Gaussian Noise on PSNR

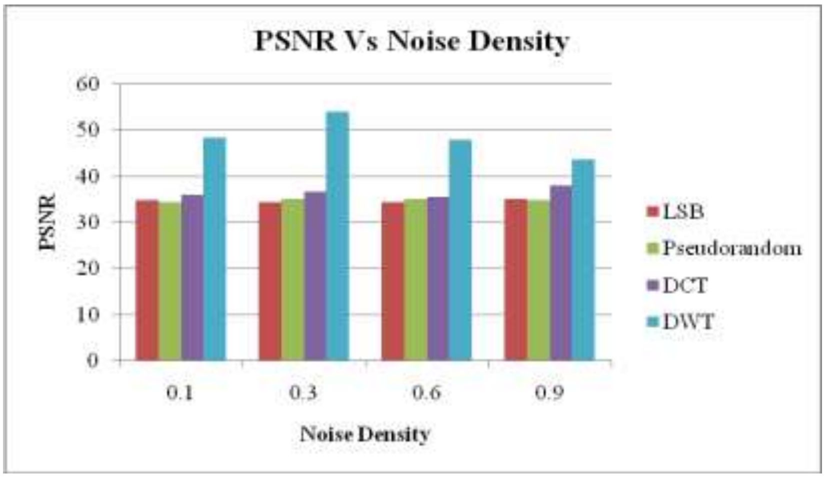


Figure 9. PSNR Vs Gaussian Noise Density

- DWT shows maximum resistance against Gaussian Noise as its PSNR values comes out to be the highest at all noise densities (see Figure 9.).
- LSB, Pseudorandom and DCT show moderate values consistently at all densities.

4.8. Impact of Salt and Pepper Noise on PSNR

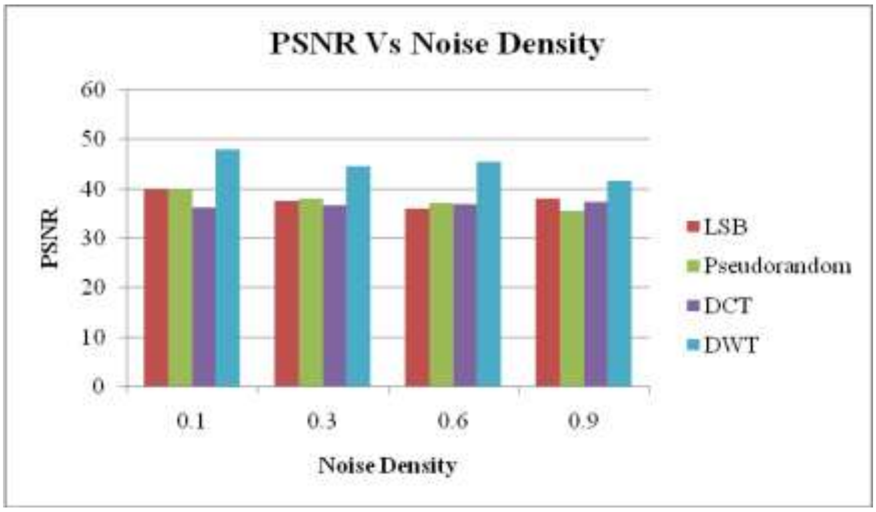


Figure 10. PSNR Vs Salt and Pepper Noise Density

- Salt and Pepper noise produces almost same effect on PSNR of all the techniques. Spatial domain methods and DCT have comparable values.
- DWT has a slight edge over other techniques, as observed from Figure 10.

4.9. Impact on Correlation Coefficient

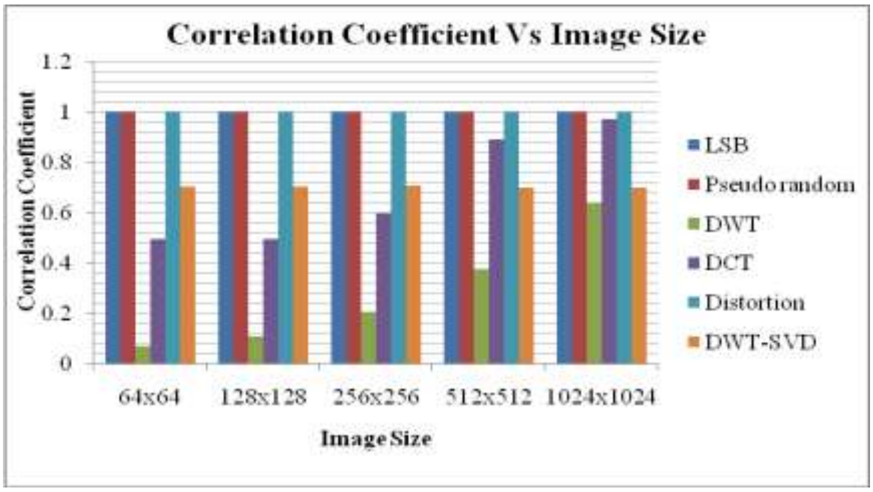


Figure 11. Correlation Coefficient Vs Image Size

- As evident from Figure 11, maximum correlation between original image and watermarked media is found in LSB, Distortion and Pseudorandom which is due to the fact that only few bits of pixel values are changed in spatial domain.
- This parameter is minimum in DWT. It can be related with the fact that image converted to greyscale and further divided into four sub-bands, reducing the correlation between cover and watermark.
- Similary, DCT and DWT-SVD also shows low correlation because of the transformation prior embedding.

4.10. Impact of Cropping on Correlation

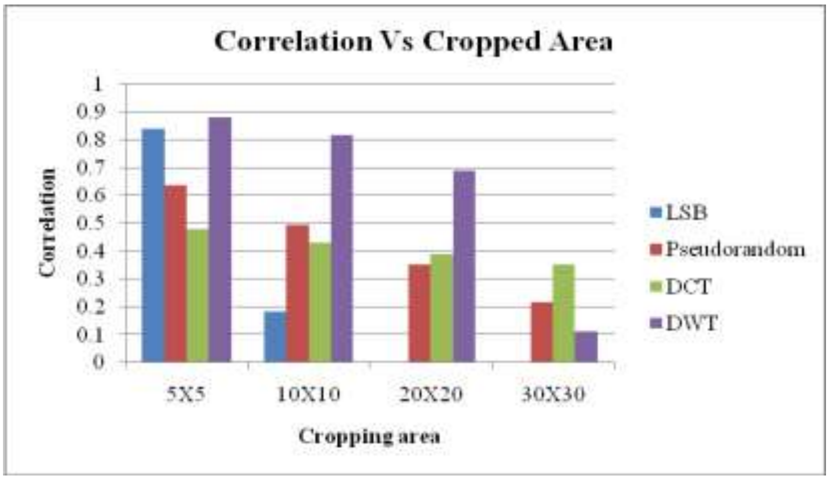


Figure 12. Correlation Coefficient Vs Cropped Area

- Correlation Coefficient is found to be highest in DWT after Cropping but when cropped area becomes large, it jumps down to lowest value.
- DCT shows very high correlation in for smaller cropped area and falls down to minimum as the area becomes large.
- Pseudorandom and DCT show moderate values (see Figure 12).

4.11. Impact of Gaussian Noise on Correlation

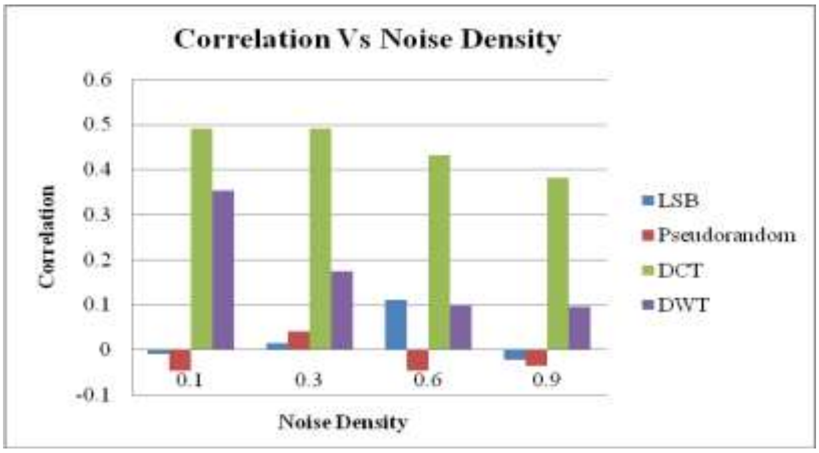


Figure 13. Correlation Coefficient Vs Gaussian Noise Density

- On applying Gaussian noise, DCT is found to maintain highest Correlation (refer Figure 13.).
- Spatial domain techniques *i.e.* Pseudorandom and LSB show extremely low values.
- DWT shows moderate correlation.

4.12. Impact of Salt and Pepper Noise on Correlation

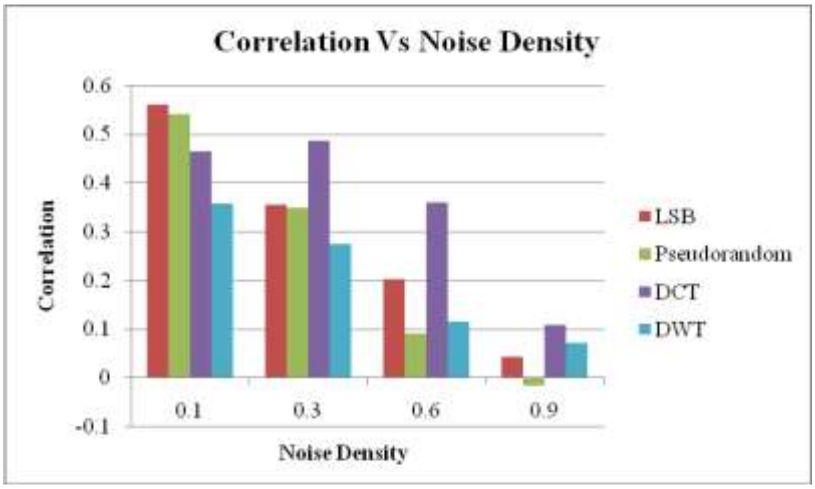


Figure 14. Correlation Coefficient Vs Salt and Pepper Noise Density

- Correlation is very high for spatial domain techniques at smaller noise density and it goes on decreasing as density increases.
- At higher noise density, correlation reduces significantly.
- DCT maintains consistently appreciable values at all noise densities (see Figure 14.).

4.13. Impact of Bit Error Rate

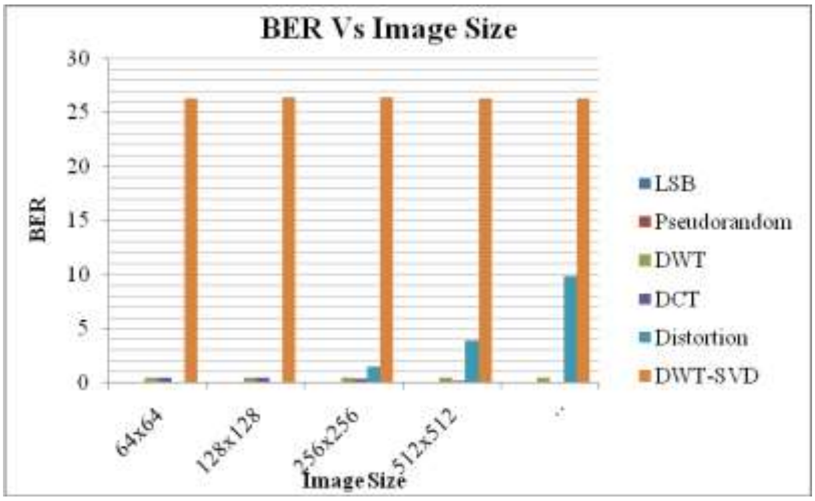


Figure 15. BER Vs Image Size

- DWT-SVD has exceptionally high Bit error rate for all image sizes.
- DCT has moderate values of BER which increases with size of image while all other techniques have much lower BER (see Figure 15.).

4.14. Impact of Cropping on BER

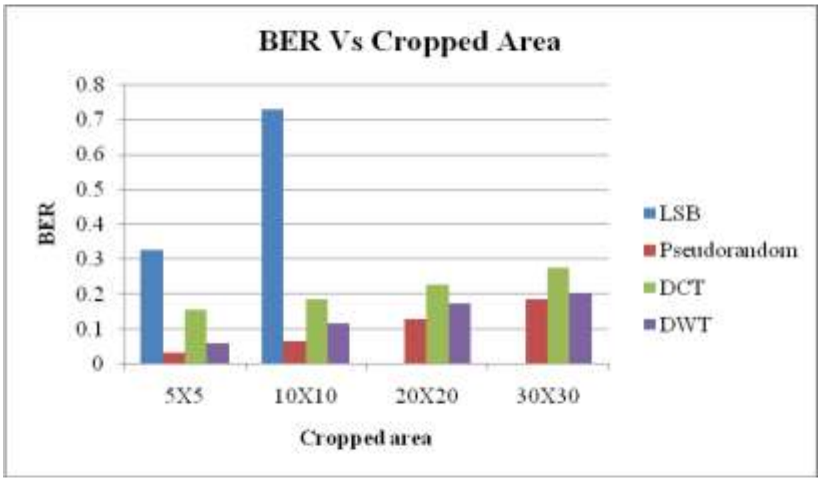


Figure 16. BER Vs Cropped Area

- When Cropping is applied, LSB shows maximum values of BER for smaller cropping area (see Figure 16.).
- DCT maintains relatively better values in other cases.

4.15. Effect of Gaussian Noise on BER

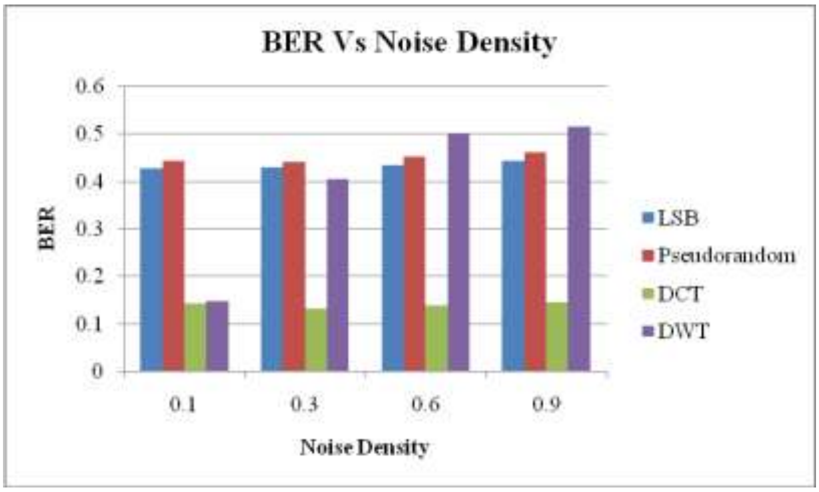


Figure 17. BER Vs Noise Density

- Except DCT, BER of all other techniques show comparably high values on application of Gaussian Noise (see Figure 17.).
- The results are found to be consistent at all noise densities.

4.16. Effect of Salt and Pepper on BER

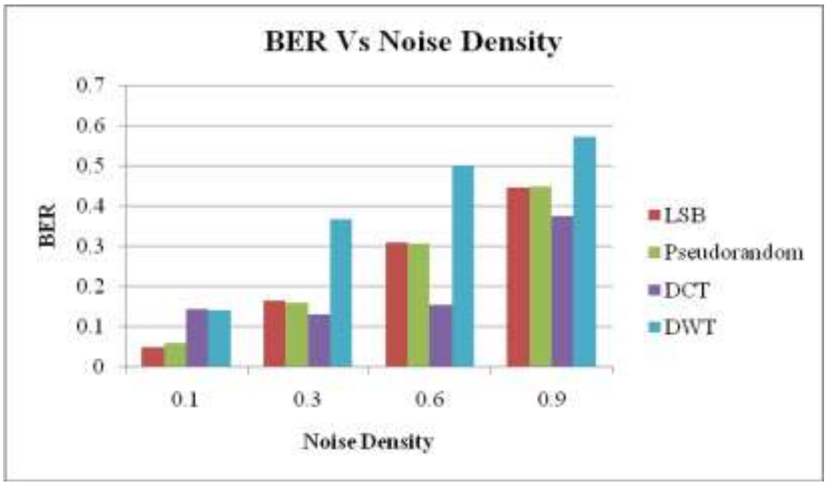


Figure 18. BER Vs Noise Density

- Bit error rate increases as the noise density increases (refer Figure 18.).
- DWT maintains the maximum value at almost every density indicating that Salt and Pepper noise doesn't have prominent impact on the BER of DWT.
- LSB and Pseudorandom techniques have moderate values of BER.
- DCT has low BER in most of the cases.

5. Conclusion

Applying the different algorithms of imperceptible watermarking, following conclusions can be drawn:

1. Frequency domain techniques are more robust against various attacks as compared to spatial domain techniques. In LSB, cropping is capable of causing loss of entire watermark information.
2. DWT Watermarking technique has highest resistance against most of the attacks and hence it is the most robust technique amongst the once compared.
3. In frequency domain techniques where image is transformed before embedding the watermark, Mean square error (MSE) is notably high as compared to other approaches.
4. As a consequence, PSNR for DCT is lowest whereas it has appreciable values for DWT, LSB, Pseudo-random.
5. Correlation is found to be maximum in spatial domain methods *i.e.* LSB and Pseudorandom. For smaller image sizes, correlation between the cover and DWT watermarked image is quite low, though it becomes comparable as the size of image increases. For DCT it is least.
6. BER is exceptionally high in DWT-SVD combination technique.
7. Image quality and Imperceptibility are preserved better in spatial domain techniques.

References

[1] R. Islam, A. Siddiqa, P. Uddin, A. Kumar Mandal and D. Hossain, “An Efficient Filtering Based Approach Improving LSB Image Steganography using Status Bit along with AES Cryptography”, in International Conference on Informatics, Electronics and Vision, Dhaka, ICIEV, **(2014)** , pp. 1-6.

[2] H. Shesi, J. Mesgarian and M. Rahmani, “Steganography: Dct Coefficient Replacement Method and Compare With JSteg Algorithm”, in International Journal of Computer and Electrical Engineering, IJCEE, vol. 4, no. 4, **(2012)**.

[3] P.-Y. Chen and H.-J. Lin, “A DWT Based Approach for Image Steganography”, in International Journal of Applied Science and Engineering, Int. J Appl. Sci, Eng., vol.4, no.3, **(2006)**.

[4] N. Tayal, S. Dhall and S. Gupta, “A robust hybrid Steganography mechanism for security in data communication networks”, in International Jounal of Computer Networks and Application, IJCNA, vol. 3, no. 3, **(2016)**.

[5] J. Zhao, N. Zhang, J. Jia and H. Wang, “Digital watermarking algorithm based on scale-invariant feature regions in non-subsampled contourlet transform domain”, in Journal of Systems Engineering and Electronics, JSEE, vol. 26, no. 6, **(2015)**, pp.1309– 1314.

[6] T. Xin, Z. Ma, X. Niu and Y. Yang, “Compressive Sensing-Based Audio Semi-fragile Zero-Watermarking Algorithm”, in Chinese Journal of Electronics, vol.24, no.3, **(2015)**, pp. 492-497.

[7] Y. Yang , M. Lei 1, 2, X. Liu, Z. Qu and C.Wang, “Novel Zero-Watermarking Scheme Based on DWT DCT”, in Information Security Center, China Communications, vol. 13, no. 7126, **(2016)**, pp. 122-.

[8] N. M. Makbol, B. E. Khoo and T.H. Rassem, “Block-based discrete wavelet transform singular value decomposition image watermarking scheme using human visual system characteristics”, in The Institution of Engineering and Technology, IET Image Process, IET, vol. 10, no. 1, **(2016)**, pp. 34-52.

[9] V.M. Potdar, S.Han and E.Chang, “A survey of digital image watermarking techniques”, in 3rd IEEE International conference on Industrial Informatics, **(2005)**, pp. 709-716.

[10] R. Liu and T.Tan, “An SVD-based watermarking scheme for protecting rightful ownership”, in IEEE Trans. Multimedia, vol. 4, no.1, **(2001)**, pp. 121-128.

[11] J. Liu and X. He, “A review study on digital watermarking”, in 1st International Conference on Information and Communication Technologies, **(2005)**, pp. 337-341.

[12] S. Malshe(Gondhalekar), H. Gupta and S. Mandloi, “Survey of Digital Image Watermarking Techniques to achieve Robustness”, International Journal of Computer Applications, vol. 45, no.13, **(2012)**, pp. 0975 – 8887.

[13] H. Tian, Y. Xiao, G. Cao, J. Ding and B. Ou, “Robust Watermarking of mobile video resistant against barrel distortion”, China Communication, vol. 13, no.19, **(2016)**.

[14] X. Tang, Z. Ma, X. Niu and Y. Yang, “Compressive Sensing-Based Audio Semi-fragile Zero-Watermarking Algorithm”, Chinese Journal of Electronics, vol. 24, no.3, **(2015)**.

[15] Y. S. Singh, B. P. Devi and K. M. Singh, “A Review of Different Techniques on Digital Image Watermarking Scheme”, in International Journal of Engineering Research, IJER, vol.2, no.3, **(2013)**, pp. 193-199.

[16] F. Daraee and S. Mozaffari, “Watermarking in binary document images using fractal codes”, Pattern Recognition Letter, vol.35, **(2013)**, pp. 120-129.

[17] A. Bamatraf, R. Ibrahim and M.N. B. Mohd Salleh, “Digital Watermarking Algorithm Using LSB”, in International Conference on Computer Applications and Industrial Electronics, ICCAIE, **(2010)**, pp. 150-159.

[18] J. Hossain, “Information-hiding using Image Steganography with Pseudorandom Permutation”, in Bangladesh Research Publications Journal,BRP, vol.9, no.3, **(2014)**, pp. 215-225.

[19] M. T. Sandford, J. N. Bradley and T. G. Handel, “Data Embedding Method”, in proceedings of the SPIE 2615, Integration Issues in Large Commercial MediaDeliverySystems, **(1996)**, pp. 226-259.

[20] N. F. Johnson and S. Jajodia, “Exploring Steganography Seeing the Unseen”, IEEE Computer, vol. 31, no. 2, **(1998)**, pp. 26-34.

[21] R.O. Preda and D.N. Vizireanu, “Watermarking-based image authentication robust to JPEG compression”, in Electronics Letters,vol. 51, no. 23, **(2015)**, pp. 1873–1875.

[22] R. Sheth and V. V. Nath, “Implementation of Digital image watermarking with the comparison between Least Significant Bit and Discrete Cosine Transform method”, in International Journal of Trend in Research and Development, IJTRD, vol. 2, no.5, **(2013)**.

[23] R. K. Sheth and V V Nath, “Secured Digital Image Watermarking with Discrete Cosine Transform and Discrete Wavelet Transform method”, in International Conference on Advances in Computing, Communication and Automation, ICACCA, **(2016)**, pp. 1-5.

[24] B. Gupta Banik and S. K. Bandyopadhyay, “A DWT Method for Image Steganography”, in International Journal of Advanced Research in Computer Science and Software Engineering, IJARCSSE, vol. 3, no. 6, **(2013)**.

[25] S. Bajracharya and R. Koju, “An Improved DWT-SVD Based Robust Digital Image Watermarking for Color Image”, International Journal of Engineering and Manufacturing, IJEM, vol.7, no.1, **(2017)**, pp.49-59.