



Robust and Selective Watermarking using DWT, SVD and Image Depth Map

¹Gaurav Kumar, ²Premkishor Gautam, ³Manoj Kumar Sharma

¹M.Tech Scholar, ²Assistant Professor, ³Assistant Professor

¹Department of Computer Science and Engineering

¹MIT Bulandshahr, U.P, India

Abstract:- The spread of digital and media content has increased the risk of copyright violations. Therefore, there is a need for a technique that can either identify or prevent copyright violations. Therefore, there is a need for a technique that can either identify or prevent copyright infringement. For this reason, either cryptographic or non-cryptographic methods can be employed, but the simplest and cheapest option is non-cryptographic, which anyone can use. Therefore, watermarking or steganography may be used. Watermarking is the concealing of digital content into the image to be watermarked. Watermark employs several embedding techniques, including DWT, DCT, SVD, etc. Both DWT and SVD have been utilised in numerous watermarking techniques that have yielded positive results and repelled numerous attacks. In numerous studies, DWT has achieved superior performance. The watermark image utilised DWT and SVD in this research. The watermarked image is processed using DWT and then divided into smaller parts. The blocks are then processed using SVD, and the watermark's binary pixels are inserted on the greatest values of singular values. The alpha values determine the durability of the watermark and the image. The experimental results of the suggested method indicate an improvement in the perceived attributes of the watermarked image, as revealed by their analysis. It was examined against attacks such as AWGN, Median Filters, etc.

Index Term: - Cryptography, Watermarking, Discrete Wavelet Transform (DWT), Singular Value Decomposition (SVD), Discrete Cosine Transform (DCT).

I. INTRODUCTION

A watermark is a distinguishable image on paper that appear as different shades of brightness or darkness when can be viewed by transmitted light and is generated by changes in the density of the material [1]. To prevent duplication, watermarks have been placed to postal stamps, banknotes, and other official papers.

1.1 Introduction to Watermarking

Like traditional watermark, digital watermarking are only discernible under specific condition, that is, when an algorithm is applied, and are otherwise unnoticeable. [2] A digital watermarking that alters the carrier signal to the point where it is discernible is useless. In digital watermarking, the signal can be audio, photographs, videos, texts, or 3D models. A signal that may simultaneously carry many watermarks. Digital watermarks do not alter the size of carrier signals, unlike metadata that is added to the carrier signal.

The required characteristics of a digital watermarking depend on the application. For copyright marking of media file, a digital watermarking must be quite strong against adjustment that can be made to the carrier signals. So that integrity must be maintained, a fragile watermarking should be utilised.

Both steganography and digital watermarking use steganographic techniques to conceal information inside noisy signals. In contrast to steganography, digital watermarking attempts to manage robustness as a primary priority usually. As digital copies of data are identical to the original, digital watermarking is a passive security measure. It only labels data but does not degrade it or restrict access to it. Digital watermarking can be used for source tracking. A watermark is an image that is placed into a digital signal at each distribution point. If a copy of the work is later discovered, from the copy watermark can be extracted and the source of distribution can be determined. Reportedly, this technology has been used to identify the source of illegally pirated movies. Watermarking is the process of embedding the author's identity into digital data so that it can be used to verify the claim when necessary.

Several ways, such as the company's logo, the author's banner, or some sort of signature, can be used to generate a watermark, which can also be created for every image [9]. This is not the primary issue; the embedding and extraction of the watermark is the primary concern. These are the primary means by which the author can assert ownership of an image. The majority of image watermarking research has thus far been conducted in these areas; the watermark remains unaltered even if the watermarked image has experienced significant distortions or alterations.

1.2 Types of Watermarking

Watermarking techniques can be categorised based on a variety of variables. Watermarking can be classified according to the following characteristics:

1. *Domain Based*
2. *Document Based*
3. *Visibility based*

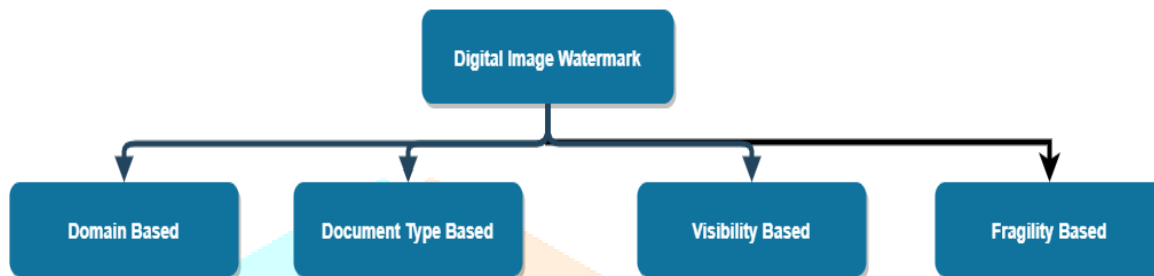


Figure 1 Bases of Watermark Categorized

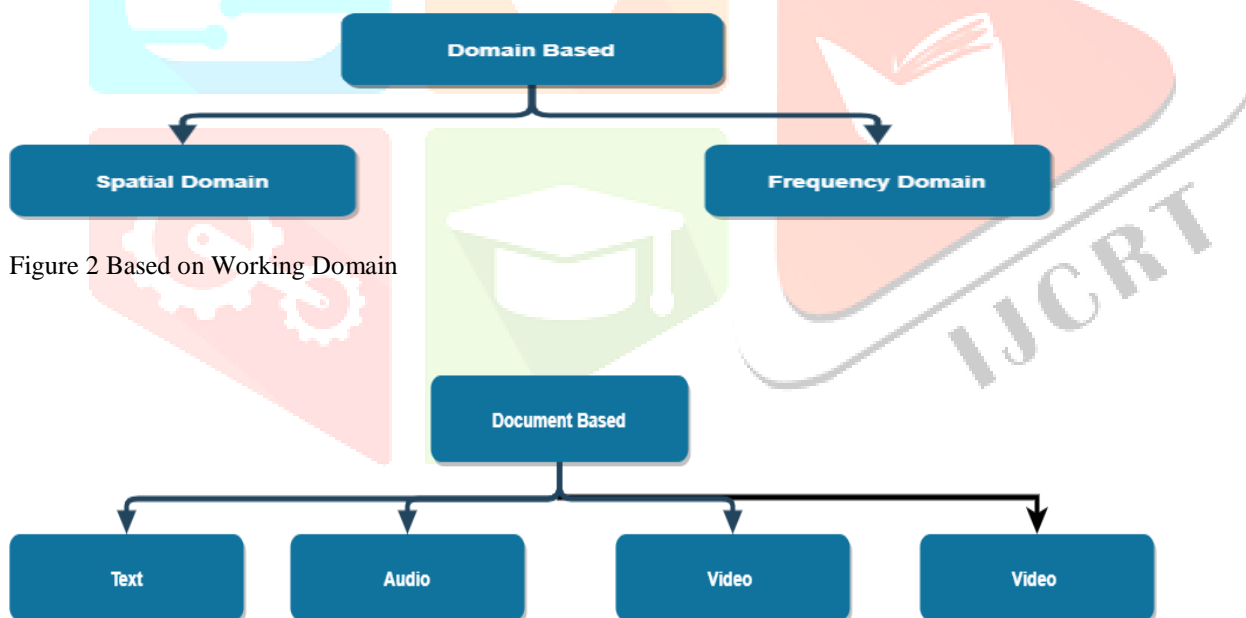


Figure 2 Based on Working Domain

Figure 3 Based on Type of Document

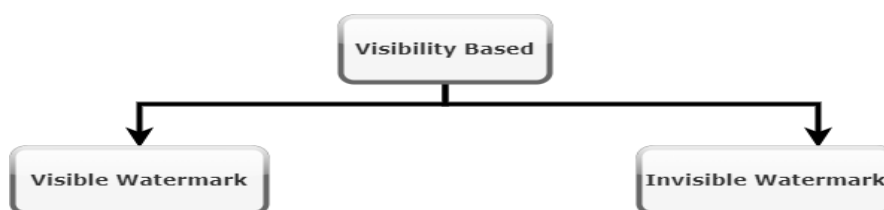


Figure 4 Based on Visibility

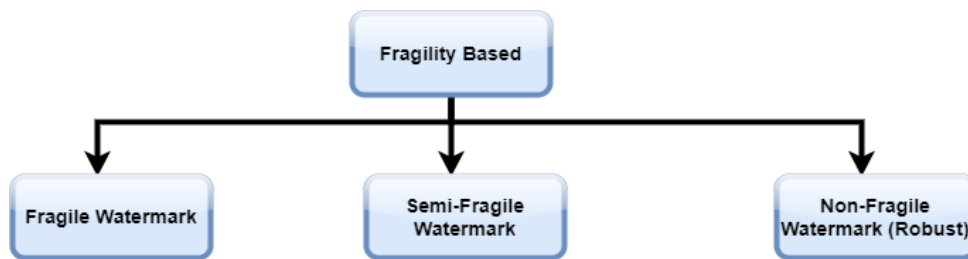


Figure 5 Based on Fragility

1.3 Characteristics of Watermark

The following is a list of characteristics [14] of watermarking:

Imperceptibility: This indicates that the watermark is present in the image, but it cannot be seen by the naked eye due to the way the image was processed.

Robustness: It is utilised in situations in which the watermark has been subjected to various image processing operations such as cropping, resizing, blurring, and so on.

Capacity: The quantity of watermark that an image can retain or that an algorithm can implant without significantly affecting the original image is referred to as the watermark capacity.

Security: A safe and secure watermark is one that cannot be located or removed by an adversary even if they had the key.

False-positive: It is the prerequisite that the detector must indicate the presence of a watermark, even though one does not exist.

1.4 Application of Watermark

Watermarking can be used to authenticate nearly all types of documents. There are numerous applications for watermarking; hence the following are some of the applications [14] for watermarking:

Information Security

Copyright Protection

Content Identification and Management

Content/object authentication

Others application

1.5 Objectives

The Objectives of the proposed work are:

Develop a watermarking algorithm that can embed the watermark adaptively, i.e., with various strengths for the background and foreground, while staying undetectable in the image.

Achieve a high PSNR and SSIM for watermarked images compared to the host image.

To accurately extract the watermark image from the watermarked image.

Apply image processing attacks and determine if the system can withstand these attacks while maintaining the integrity of the watermark, and if so, to what extent.

II. LITERATURE SURVEY

The important qualities of a watermarking algorithm [32] are its fragility, visibility, and capacity. A watermark, which is usually dealt with, should be robust, invisible, and have a large capacity. It should be noted that if there is a need for the validity check of an image, then the watermark should be fragile and invisible. However, for the authenticity of the image, several other methods are used. Our major aim for this survey is to search for robust and invisible watermarks. It should be noted that capacity [34] is usually not important because only a small amount of information, like images, text, etc., is added as a watermark. It is not too important to consider it as a part of the survey. As previously stated, spatial domain watermarks are lower resistant to image processing modifications. The transform domain [36] must be included for the survey. It improves the capacity of watermark placement in images **Error! Reference source not found.**

Burgett et al. introduced the first efficient watermarking technique in the DCT domain [23]. Their approach partitions the image into 8x8 blocks for which the DCT is figure out. In each block, a pair of coefficients is selected. To embed a bit, the coefficients are swapped so that, depending on the bit value, the value of positional difference is either positive or negative.

The method described in **Error! Reference source not found.** models the watermarking channel as a generalised channel with fading and mean additive noise that is not zero. Utilizing an adaptive extractor, they present an optimised watermark extraction approach for their quantization-based watermarking. In [20], four DWT levels are selected as the watermarking domain. This is settled to strengthen the approach while concealing the watermark.

The technique of watermarking in short frequency coefficients of the host image using a sample projection approach is analysed in [21].

The algorithm described in [35] embeds the watermarking in JPEG2000-compressed and encrypted image. This approach utilises the stream cypher technology during the encryption phase. A quantize watermarking technique based on feature modulation was developed in[38]. In this method, the feature signal is the normalised correlation (NC) between the host signal and a certain random signal. In the [31] literature, many embedding, and extraction techniques have been investigated. In [24][25], for instance, GA is used for watermarking purposes. One of the most important applications of watermarking is the detection of forgeries. In [28][37], watermarking is utilised to detect tampering in various multimedia settings. The paper [29] proposes a watermarking approach based on CT. The watermark is embedded in the DCT coefficients of CT's detail blocks. They are inserted in complicated blocks with a greater strength factor. This is done to achieve greater durability. Also, to achieve imperceptibility, smooth blocks are embedded with a weaker force. Using edge concentration and block entropy, the difficulty of each detailed block is determined.

III. PROPOSED APPROACH

Watermarking involves two sub-procedures: embedding the watermark and extracting the embedded watermark. The watermark embedding module includes inserting the watermark into the host image by modifying the intermediate values of the host image part (singular values) with the watermark part (watermark pixels). For controlling the strength of the watermark and the image, another parameter called image depth maps (here it is ground truth images) is used.

In the extraction of watermarking module, the watermark that was inserted in the embedding module is taken out of the watermarked image. For that, the procedure is just the reverse of that embedding.

Some basic terminology used in the watermarking process is as follows:

Host Image: It is the image on which the watermark must be applied. This is a color RGB image.

Watermark Image / Content: This is the data which must be embedded in the original host image to make it a watermarked image.

Watermarked Image: It is an image with watermark content embedded into it. It is obtained after applying a watermarking process.

Ground-Truth Image: It is an image that is obtained after applying Image Depth Map techniques. The Image Depth Map is a separate topic of interest. This would be used to separate the foreground and background images of the host image according to object interest.

3.1. Transformation Techniques

1. *Singular Value Decomposition*:- SVD is a technique for linear transformation in algebra. SVD [4] is the decomposition of a given matrix into two orthogonal matrices and a diagonal matrix. If X is a $m \times n$ matrix that is to be decomposed using SVD, then U and V are orthogonal matrices of dimensions $m \times m$ and $n \times n$, respectively, and Σ is a diagonal matrix with the dimensions $m \times n$, same as X .

$$X = U\Sigma V \dots \dots \dots (1)$$

Σ is known as the singular values (SVs) matrix in equation (1). can have many zeroes at the bottom and rightmost corners of the diagonal matrix. Left singular vectors (LSC) and right singular vectors (RSC) are U and V , respectively. The LSC of X is derived from eigenvectors of XX^T and the RSC of X is derived from eigenvectors of $X^T X$. SVs of the image represent the brightness, whereas singular vectors (SCs) determine the image's geometry. [5]

2. *Discrete Wavelet Transform (DWT)*:- It is extensively utilised in the domains of signal processing and image processing. DWT is a multi-resolution analysis that decomposes an image into fixed-bandwidth frequency channels on a logarithmic scale. One of the benefits of DWT is its resemblance to a structure according to a particular resolution and decomposition at each level

3. *Blocking*:- Blocking is the division of a huge matrix into smaller matrices. Typically, a large matrix is partitioned into 4x4 blocks. Before dividing, the compatibility of the matrix for division is examined. If not, compatibility is achieved by rounding up to the nearest multiple of four. After division, the two dimensions are preserved and can be utilised as necessary.

4. *Alpha Selection*:- The Image Depth map provides ground-level photographs. Consequently, for the alpha selection technique, the ground truth image would be required to depict the foreground and background twenty-six portions of the image. The empirically determined effective foreground and background alpha values are 0.96 and 0.98 for the foreground and background, respectively. The ground truth image is initially binarized using well-known thresholding techniques, such as OSTU global thresholding.

5. *Watermark Intermediate Value Modification*:- To embed the watermark into the host image, certain prerequisite values must be established in the preceding phases. The host image is pre-processed to obtain a group of 4x4 blocks containing the greatest single values from each of the 4x4 input blocks. These singular values are derived from the SVD procedure described in above techniques.

$$A = \alpha B + (1 - \alpha)t \dots \dots \dots (1)$$

B is a component of the host image following pre-processing. α is the image part's strength factor utilised in the watermarked image. t is the portion of the watermark image to be utilised in the watermarked image. This equation is also known as alpha blending. [12][13][15][16].

3.2 Watermark Techniques

The Watermark Embedding Module embeds the watermark into the image. The implementation is described in depth. The procedure entails separating the RGB components and applying the watermark to each of them, i.e., red, green, and blue. The watermark in these components will be utilised to determine whether version of the watermark extracted from the watermarked image being processed here is the most accurate. This describes the input, output, and overall operation of the watermark embedding process.

INPUT:

Host Image: Let us say H1.png

Watermark Image: Let it be W1.png

Ground-truth Image: For Depth Map parameter. Let it be G1.png

OUTPUT:

Watermarked Image: After applying watermark in Host Image.

Measuring Parameter Value: PSNR and SSIM

ALGORITHM:

Initialization Of the Input Variables

Read H1.png using imread () function in MATLAB. Store it in variable, say imgH1.

Read W1.png and store it in a variable, wimg1.

Read GT1.png and store it in a variable, gtimg1

imgH1, wimg1, and gtimg1 are all 2D arrays of the pixel value.

2. Host Image Pre-Processing

Split the RGB components of image imgH1 and proceed for each component the underneath steps. Let it be *redComp*, *greenComp* and *blueComp*.

Apply Discrete Wavelet Transform (DWT) level-1 on imgH1, to get four components *cA*, *cH*, *cV* and *cD*.

$$[cA, cH, cV, cD] = dwt2(imgH1, 'Haar')$$

"*cA*" is an approximate image, not exactly the copy of original one, it looks but it is not. "*cH*" is the horizontal details of the image; "*cV*" is the vertical details of the image; "*cD*" is the diagonal details. "dwt2()" is a built-in MATLAB function for 2-D DWT transform.

Now *cA* is selected for the further processing.

Divide the *cA* into 4×4 . It is stored in block4x4. A point to be noted is that block4x4 is a 3D array, multiple 4×4 blocks are stored in layers one after another.

Apply singular value decomposition (SVD) on the block4x4.

$$[u_{comp}, s_{comp}, v_{comp}, singularValue] = svdTransform(block4x4)$$

u_{comp} is a 4x4 orthogonal matrix (3D array), s_{comp} is 4x4 diagonal matrix (3D array), v_{comp} is a four-by-four orthogonal matrix (3D array). These have multiple entries against each four-by-four entry of block4x4. *singularValue* is 3D array of dimension $4 \times 4 \times M$, with entries having first elements of four-by-four of s_{comp} . *svdTransform()* is a user-defined function which applies the SVD on each of the 4x4 block of *block4x4* using a built-in MATLAB function *svd()* giving the output as mentioned above.

3. Ground Truth Image Pre-Processing

Apply DWT on gtimg1.

$$gtcA, gtcH, gtcV, gtcD = dwt2(gtimg1, 'Haar')$$

b) Convert *gtcA* to binary image

$$bwImage = imbinarize(gtcA);$$

"*bwImage*" is matrix having entries 0 and 1.

c) Each entry of one is replaced by the alpha value of the foreground, and zero is replaced by the alpha value of the background. The α value of the foreground and background must be selected experimentally. It is the strength factor of the host image part used in watermarked images, and $(1 - \alpha)$ is the strength factor of the watermark part. Applying these operations, in the end, we get a 2-D array.

d) Blocking of *alpha2DArray* to 4×4 blocks

$$alpha3DArray = blocking4x4fromnxdn(alpha2DArray)$$

4. Watermark Image Pre-Processing

Adjust the size of the watermark image to match the size of *singularValue*.

$$sz1 = size(singularValue)$$

$$dimensionsOfWatermark = \left\lceil \sqrt{sz1(1) * sz1(2) * sz1(3)} \right\rceil \left\lceil \sqrt{sz1(1) * sz1(2) * sz1(3)} \right\rceil$$

$$watermarkResized = imresize(wimg1, dimensionsOfWatermark)$$

watermarkResized is resized version of watermark. Its dimension is matching with *dimensionsOfWatermark*

Division of the *watermarkResized* into 4x4 blocks

$$blockedWatermark4x4 = blocking4x4fromn \times n(watermarkResized)$$

blockedWatermark4x4 is a four-by-four blocked watermark image.

DCT is applied on *blockedWatermark4x4* to get *dctBlockedValue*

$$dctBlockedValue = dctBox(blockedWatermark4x4)$$

dctBlockedValue is DCT output of 4x4 blocks in the previous steps.

5. Watermark Intermediate Value Modification

In this section, value modifications are made. *singularValue* from host image pre-processing, *dctBlockedValue* from watermark pre-processing, and *alpha3DArray* from GT Image preparation are the values used. The operation listed below is used to change *singularValue*. It is applied to each (i, j) array element.

$$modSingularValues4x4 = singularValue * alpha3DArray + (1 - alpha3DArray) * dctBlockedValue$$

This is called alpha-blending.

6. Post- Processing

Retiling *singularValue4x4mod* to s_{comp} to generate modified singular values and take inverse SVD.

$$[s_{compMod}, imgblock4x4] = insvd(singularValue4x4mod, u_{comp}, s_{comp}, v_{comp})$$

Retile *imgblock4x4* blocks to form an approximate image i.e., *cAmod*.

Apply Inverse DWT to get final watermarked image.

$$wdimgFinal = idwt2(cAmod, cH, cV, cD, 'Haar')$$

cAmod is a modified approximate image *cH*, *cV* and *cD* are the values obtained in step 1.

7. Output

wdimgFinal is written to an image file. It is compared with Host Image H1.png, it should be visibly same. So, its PSNR values and SSIM values are recorded to compare how similar they are.

3.3 Watermark Extraction

INPUT:

Host Image: Let us say H1.png

Watermark Image: Let it be W1.png

Ground-truth Image: For Depth Map parameter. Let it be G1.png

OUTPUT:

Watermark Image: The watermark image which was embedded into the host image.

Measuring Parameter Value: PSNR and SSIM, to measure the accuracy of watermark extraction.

FULL ALGORITHM:

1. Initialization Of Input Variables

imgH = read H1.png, Host-image

gting= read GT1.png, ground-truth image

wmdimg=read wimg1File.tiff, watermarked image.

wimg = read w1.png, watermark.

2. Host Image Pre-Processing

Split the RGB components of image imgH1 and proceed for each component the underneath steps. Let it be redComp, greenComp and blueComp.

Apply Discrete Wavelet Transform (DWT) level-1 on imgH1, to get four components cA, cH, cV and cD.

$$[cA, cH, cV, cD] = dwt2(imgH1, 'Haar')$$

cA is an approximate image, cV is the vertical details of the image; cD is the diagonal details. *dwt2()* is a built-in MATLAB function for 2D DWT transform.

Now cA is selected for the further processing.

Divide the cA into 4×4 blocks. It is stored in block4x4. A point to be noted is that block4x4 is a 3D array, multiple 4×4 blocks are stored in layers one after another.

Apply singular value decomposition (SVD) on the block4x4

$$[u_{comp}, s_{comp}, v_{comp}, singularValue] = svdTransform(block4x4)$$

3. Ground Truth Image Pre-Processing

Apply DWT on gting1.

$$gtcA, gtcH, gtcV, gtcD = dwt2(gting1, 'Haar')$$

Convert gtcA to binary image

$$bwImage = imbinarize(gtcA);$$

bwImage is matrix having entries 0 and 1.

Blocking of *alpha2DArray* to 4×4 blocks

$$alpha3DArray = blocking4x4fromnxdn(alpha2DArray)$$

4. Watermarked Image Pre-Processing

Apply DWT on wmdimg

$$[wcA, wcH, wcV, wcD] = dwt2(wmdimg, 'Haar');$$

Select wcA for further processing

Divide wcA into 4×4 blocks.

$$wimgblock4x4 = blocking4x4fromnxdn(wcA)$$

Apply SVD on *wimgblock4x4*

$$[wu_{comp}, ws_{comp}, wv_{comp}, wsingValue] = svdTransform(wimgblock4x4)$$

5. Watermark Value Detection

$$extWImg4x4 = \frac{wsingValue - singValue * alpha3Darray}{1 - alpha3Darray}$$

In this part, the extraction of the watermark from the watermarked image is done. For applying the operation, we would need value *wsingValue* from the Watermarked Image pre-processing step, *alpha3Darray* from the GT Image pre-processing, *singValue* from Host Image pre-processing. These are used to calculate the watermarked content using the formula

6. Post-Processing

extWImg4x4 is a DCT value of watermark extracted, it is need to applied inverse DCT on *extWImg4x4* to get a reconstructed *invDCTBlocked*.

Since *invDCTBlocked* is in blocked form it is needed, to be retiled into an image. So, after retiling the blocks into image we get the watermarked image.

$$extWatermarkImage = retiling4x4intoImage(invDCTBlocked, dimensionofWimg)$$

7. Output

extWatermarkImage is a watermark extracted image. It is exactly same visually, so for measuring the similarity of the two images, MSE and SSIM is calculated. For binary watermark, the 1s and 0s are replaced by a minimum threshold e.g., 251 and 53, respectively. After that, all the steps are same.

IV. RESULT AND ANALYSIS

In this part, the algorithm for watermarking has been evaluated and the measured values of various parameters have been recorded. The Peak-Signal-to-Noise-Ratio (PSNR) and the Structural Similarity Index (SSI) are the criteria used to analyse the watermarked images (SSIM). SSIM and Mean-Squared Error (MSE) are more often employed to evaluate the retrieved watermark image. However, Bit-Error Rate (BER) can also be employed, but it is more appropriate for binary data. . The foreground and background alpha values are set at 0.96 and 0.98, respectively. The watermark picture has a size of 64 by 64 pixels, whereas the host image has proportions of 512 by 512 pixels. A grayscale and binary watermark have been applied. Additionally, the size of GT images is the same as the size of the host image.

$$MSE = \frac{1}{\sum_{i=1}^m \sum_{j=1}^n} [x(i, j) - y(i, j)]^2 \quad (1)$$

$$PSNR = 10 \log_{10} \frac{\max(x(i, j))^2}{MSE} \quad (2)$$

$$SSIM(x, y) = \frac{(2\mu_x + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (3)$$

$$BER = \frac{\sum_{i=1}^m \sum_{j=1}^n x_{ij} - y_{ij}}{\text{total number bits}} \quad (4)$$

TABLE 4.1 ANALYSIS OF WATERMARK EMBEDDING MODULE

S.No	Image File Name	PSNR	SSIM
1	Baby3.png	49.27	0.9998
2	Bowling.png	47.34	0.9997
3	Bowling2.png	44.14	0.9991
4	Cloth1.png	48.86	0.9998
5	Cloth2.png	48.19	0.9999
6	Cloth3.png	45.38	0.9998
7	Cloth4.png	47.67	0.9997

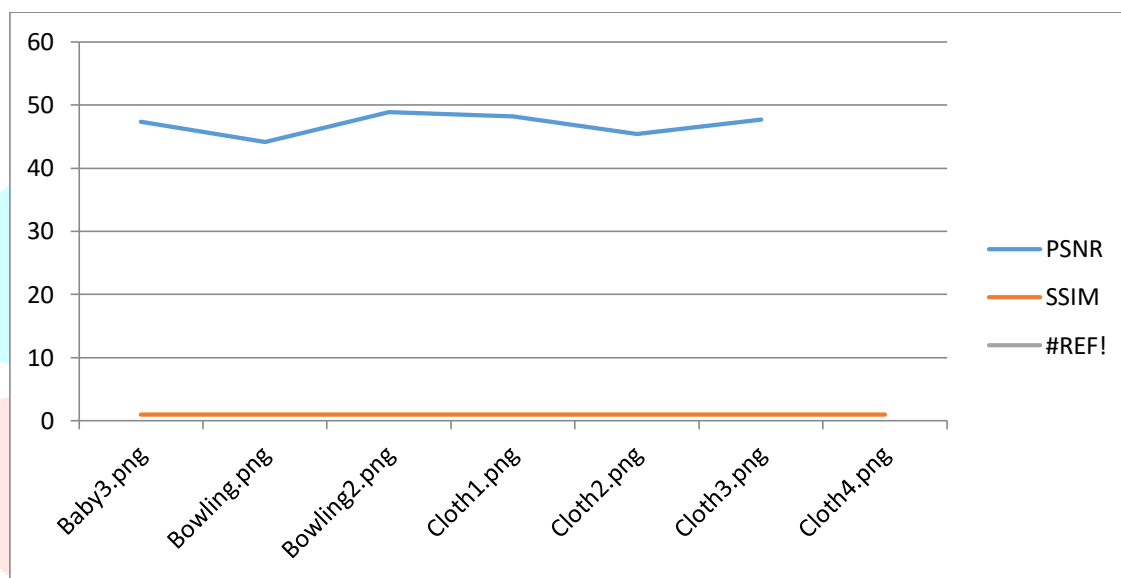


Figure 6 PSNR and SSIM Values

The PSNR values are exceptionally high, indicating that the signal-to-noise ratio is exceedingly high. The SSIM values are close to 1, indicating that the watermarked image is nearly identical to the original image. The average degree of resemblance is 99.48 percent, making it the same image.

TABLE 4.2 ANALYSIS OF WATERMARK EXTRACTION MODULE

S.No	Image File Name	SSIM	MSE
1	Baby3.png	1	0
2	Bowling.png	1	0
3	Bowling2.png	1	0
4	Cloth1.png	1	0
5	Cloth2.png	1	0
6	Cloth3.png	1	0
7	Cloth4.png	1	0

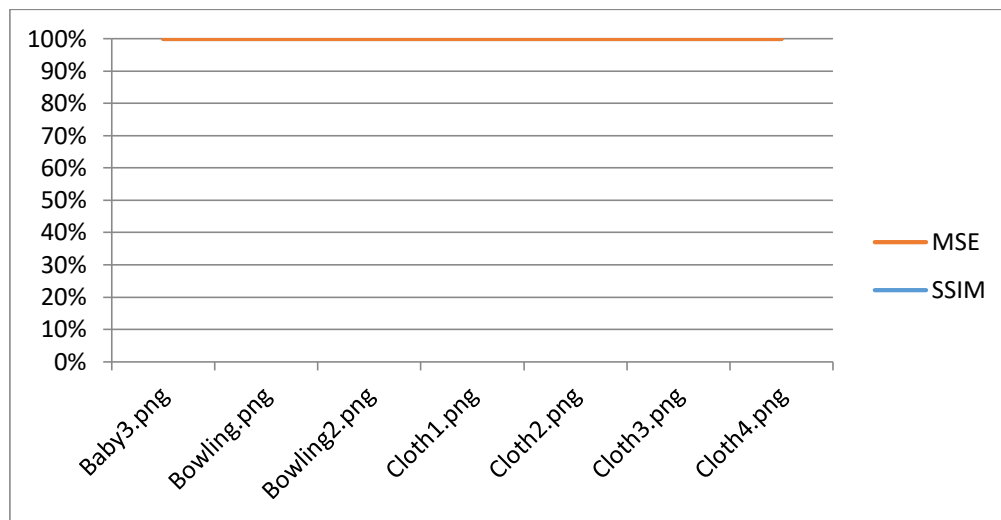


Figure 7 SSIM and MSE Values

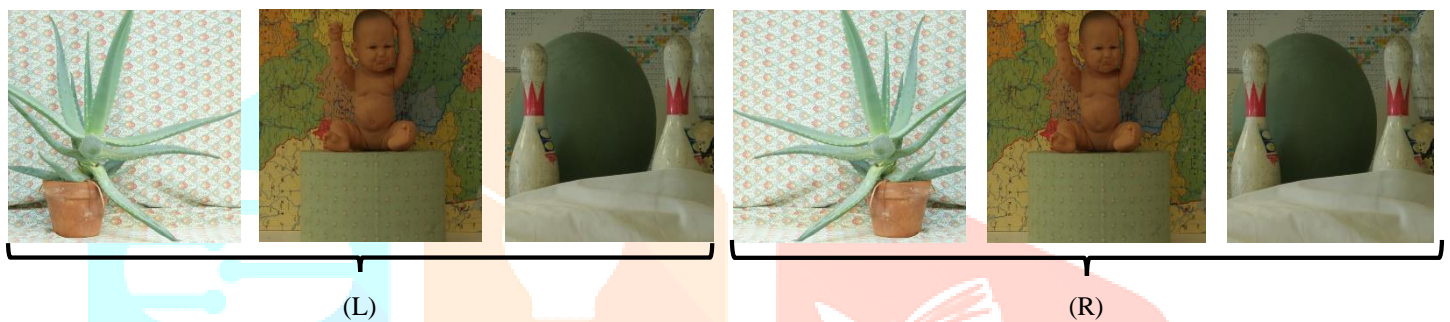


Figure 8 Image (L) Before Watermark and Image (R) After Watermark

V. CONCLUSION AND FUTURE SCOPE

The goal of this project was to design an algorithm that can insert a watermark into an image by using image processing tools like DWT and SVD. The watermark should have been invisible to the human visual system. The binary watermark was taken into account for the watermarking process. In addition to that, RGB images were chosen for this process. In the previous chapter, we have seen that the method is working positively for both processes, i.e., forward, and reversed process. By forward, we mean embedding, and reverse means extracting the watermark image. We applied the image processing attack on the watermarked images and then we tried to extract the watermark out of them. We tried AWGN, Median Filter, Scaling, and JPEG compression on the watermarked image, and then extraction was applied to find out the extent to which the watermark had been extracted from the image. To measure the values, MSE, BER, and SSIM are used. It has been observed that the algorithm performed very well with the bare images and watermarked images. However, it is not exactly similar in the case of applying DIP attacks and extracting watermarks from them. It was expected that at least the algorithm might perform very well with AWGN and the Media filter, but it is not that way. The reason behind the lack of performance as per the expectation is that while generating the image from DWT to SVD and back, the values are modified to insert the watermark, but in doing so the image pixels generated after embedding are all floating-point numbers. Also, when it is rounded-off to the nearest whole number, some of the value gets lost due to floating numbers. This problem needs to be addressed in future research.

Future scope:

Although there can be various improvements to be made to this research, some of the points to highlight for future scope are as follows:

1. The problem with lossy type casting is that it causes the watermark to be distorted initially.
2. The method can be extended to grayscale watermarks and RGB watermarks as well.
3. It has been noted that while applying the JPEG compression, the watermark gets destroyed, at least till quality factor 75%.

REFERENCES

- [1] Biermann, Christopher J. (1996). "7". Handbook of Pulping and Papermaking (2 ed.). San Diego, California, USA: Academic Press. p. 171. ISBN 0-12-097362-6.
- [2] Ingemar J. Cox, Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, Ton Kalker, "Digital Watermarking and Steganography," 2nd Edition, Morgan Kaufmann Publishers, 2008 (ISBN 978-0-12-372585-1).
- [3] Frank Y. Shih: "Digital watermarking and steganography: fundamentals and techniques," Taylor & Francis, Boca Raton, FL, USA, 2008
- [4] Dan Kalman, "A Singularly Valuable Decomposition: The SVD of a Matrix," The College Mathematics Journal, Vol. 2, No. 1, January 1996.
- [5] R. A. Sadek, "SVD Based Image Processing Applications: State of The Art, Contributions and Research Challenges," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 3 No. 7, 2012.
- [6] Xiaowei Xu, Scott D. Dexter and Ahmet M. Eskicioglu, "A hybrid scheme for encryption and watermarking. Security, Steganography, and Watermarking of Multimedia Contents," 2004, 725-736
- [7] K. Konstantinides, B. Natarajan, and G.S. Yovanof, "Noise Estimation and Filtering Using Block-Based Singular Value Decomposition," IEEE Trans. Image Processing, vol. 6, pp. 479- 483, March 1997.
- [8] E. Ganic and A. M. Eskiciogulu, "Secure DWT-SVD Domain Image Watermarking: Embedding Data in All Frequencies," ACM Multimedia and Security Workshop 2004, Magdeburg, Germany, September 20-21, 2004
- [9] P. Bandyopadhyay, S. Das, and S. Paul, "Color Image Authentication through a Dynamic Fragile Watermarking Framework," International Conference on Methods and Models in Computer Science, 2009.
- [10] H. Mohaghegh, N. Karimi, and Shadrokh Samavi, "Framework for Watermark Robustness Adjustment Using Image Depth Map," 23rd Iranian Conference on Electrical Engineering (ICEE), 2015.
- [11] C.C. Lai and C.C. Tsai, "Digital Image Watermarking using Discrete Wavelet Transform and Singular Value Decomposition," Instrum. Meas. IEEE Trans., vol. 59, no. 11, pp. 3060-3063, 2010.
- [12] N. Dey, A.B. Roy, and S. Dey, "A Novel Approach of color Image Hiding using RGB color planes and DWT," IJCA, vol. 36, No. 5, Dec. 2011.
- [13] A.P. Shing and A. Mishra, "Wavelet Based Watermarking on Digital Image," IJCSE, vol. 1, No. 2,86-91.
- [14] L. Chongmin, H. Tao, J.M. Zain and A.N. Abdalla, "Robust Image Watermarking Theories and Techniques: A review," Journal of Applied Research and Technology, Feb. 2014.
- [15] N. Kashyap and G.R. Sinha, "Image Watermarking Using 3-Level Discrete Wavelet Transform (DWT)," MECS, 2012, 3, 50-56.
- [16] B. Shen, I.K. Sethi and V. Bhaskaran, "DCT Domain Alpha Blending," IEEE, 1996
- [17] A.M. Eskicioglu, and P. Tao "A robust multiple watermarking scheme in Discrete Wavelet Transform domain," SPIE digital Library, 2004
- [18] R. Islam and Jong-Myon Kim, "Reliable RGB Color Image Watermarking using DWT and SVD," 3rd International Conference on Informatics, Electronics and Vision 2014.

- [19] , S. M. R. Soroushmehr M. Heidari, S. Samavi, S. Shirani, N. Karimi and K. Najarian, "Framework for robust blind image watermarking based on classification of attacks," *Multimed Tools Appl* DOI 10.1007/s11042-016-4150-3, Springer 2016.
- [20] Akhaee MA, Sahraeian S, Sankur B, Marvasti F, "Robust scaling-based image watermarking using maximum-likelihood decoder with optimum strength factor." *IEEE Transactions on Multimedia* 11(5):822– 833, 2009.
- [21] Akhaee MA, Sahraeian SME, Craig J, "Blind image watermarking using a sample projection approach," *IEEE Transactions on Information Forensics and Security* 6(3):883–893, 2011.
- [22] Al-Haj A, "Combined DWT-DCT digital image watermarking," *J Comput Sci* 3(9):740–746, 2007.
- [23] Burgett S, Koch E, Zhao J, "A novel method for copyright labelling digitized image data," Technical Report of Fraunhofer Institute for Computer Graphics, Darmstadt, 1994.
- [24] Chen YH, Huang HC "Coevolutionary genetic watermarking for owner identification," *Neural Computer & Application* 26(2):291–298, 2015.
- [25] Shi Y, Chu SC, Huang HC, Wu SY, Shieh CS, "Genetic watermarking for zero tree-based applications," *Circuits, Systems & Signal Processing* 27(2):171–182, 2008.
- [26] Memon N et al, Craver S, "Resolving rightful ownerships with invisible watermarking techniques: limitations, attacks, and implications," *IEEE Journal on Selected Areas in Communications* 16(4):573–586, 1998.
- [27] Communications 16(4):573–586, 1998.
- [28] Zhu YS ,Fan Y, Liu Z, "An improved SIFT-based copy-move forgery detection method using T-linkage and multi-scale analysis," *Journal of Information Hiding and Multimedia Signal Processing* 7(2): 399–408, 2016.
- [29] Karimi N, Fazlali HR, Samavi S, Shirani S, "Adaptive blind image watermarking using edge pixel concentration," *Multimedia Tools and Applications* 1:16, 2016.
- [30] Mirzakuchaki S, Hamghalam M, Akhaee MA, "Geometric modelling of the wavelet coefficients for image watermarking using optimum detector," *IET Image Process* 8(3):162–172, 2014.
- [31] Li Z, Yap KH, Lei BY, "A new blind robust image watermarking scheme in SVD-DCT composite domain," *Proc. of IEEE International Conference on Image Processing (ICIP)* 2757–2760, 2011.
- [32] Miller ML, Cox IJ, Linnartz JP, Kalker T, "A review of watermarking principles and practices," *Digital Signal Processing for Multimedia Systems*:461–485, 1999.
- [33] Nasir I, Khelifi F, Jiang J, Ipson S, "Robust image watermarking via geometrically invariant feature points and image normalisation," *IET Image Process* 6(4):354–363, 2012.
- [34] Podilchuk CI, Delp EJ, "Digital watermarking: algorithms and applications," *IEEE Magazine, Signal Processing* 18:33–46, 2001.
- [35] Subramanyam AV, Emmanuel S, Kankanhalli MS, "Robust watermarking of compressed and encrypted JPEG2000 images," *IEEE Transactions on Multimedia* 14(3):703–716, 2012.
- [36] Surekha B, Swamy GN, "A spatial domain public image watermarking," *International Journal of Security and Its Applications* 5(1):1–12, 2011.
- [37] Kim NS, Wang S, Miyauchi R, Unoki M, "Tampering detection scheme for speech signals using formant enhancement-based watermarking," *Journal of Information Hiding and Multimedia Signal Processing* 6:1264–1283, 2015.
- [38] Xinshan Z, Ding J, Dong H et al, "Normalized correlation-based quantization modulation for robust watermarking," *IEEE Transactions on Multimedia* 16(7):1888–1904, 2014.
- [39] Wikipedia 2022, "Digital Watermarking," accessed 05 June 2022, https://en.wikipedia.org/wiki/Digital_watermarking.