DOI: 10.21917/ijivp.2021.0350

A NOVEL 3-LEVEL DWT AND CNN-BASED BLIND GRAYSCALE IMAGE WATERMARKING FOR COPYRIGHT PROTECTION AGAINST ADVERSARIAL ATTACKS

Sai Shyam Sharma and Venkatachalam Chandrasekaran
Department of Mathematics and Computer Science, Sri Sathya Sai Institute of Higher Learning, India

Abstract

Copyright protection of digital images is an important commercial requirement to individual artists and large organisations alike. Wavelet-based image watermarking methods have been in practice due to their robustness against standard geometrical and image processing attacks. Convolutional Neural Networks (CNNs)-based watermarking methods are becoming popular as they provide a new dimension to the generation of a watermarked image, which is perceptually close to the original image when trained over a large class of images, thereby eliminating the need to train on each image that is to be watermarked. However, the watermark extraction performance of CNNs when used in standalone mode reduces in the presence of adversarial examples. In this study, we combine the robustness of a multi-level Discrete Wavelet Transform (DWT) and the power of CNNs and propose a robust blind grayscale image watermarking method. In the proposed method watermark is of the same size as the original image thereby demonstrating the robustness under increased payload as well. The quality of the extracted watermark is measured using Structural Similarity Index Measure (SSIM), Peak-Signal-to-Noise ratio (PSNR) and Normalized Cross Correlation (NCC). Our proposed method provides high quality watermark extraction under geometrical, image processing and adversarial attacks including second watermarking by an attacker.

Keywords:

Wavelets, CNN, Blind Watermarking, Copyright Protection

1. INTRODUCTION

Until a decade ago copyright protection was a major concern mainly for the entertainment industry but in this age of digital multimedia communication it is a serious concern for governments, corporate and even individuals. Due to the extensive personal use of social media platforms like Facebook, Instagram, Snapchat etc, the possibility of some adversary stealing information and then claiming to be the real owner through data manipulation and destruction of any in- built authentication is very high. To face this challenge, digital watermarking (visible, invisible) techniques have been used extensively for content authentication, copyright protection, tamper detection, traitor tracing, steganography etc. However, the adversaries are also working diligently to break the watermarking systems through novel attacks. This being a constant battle, researchers are placing themselves in the adversaries' role to find methods that would be robust against any futuristic attacks. Watermarking methods mainly are classified as spatial domain or frequency domain methods. In addition, the watermark extraction processes to establish legal ownership are generally classified as non-blind, semi-blind and blind techniques.

In this study, we focus on blind watermarking under frequency domain approaches. In the literature, we find the use of various frequency domain transforms and notably among them are Discrete Fourier transform (DFT), Short-time Fourier Trans- form (STFT) and Discrete Wavelet Transforms, (DWT). All three transforms are inner product transforms, meaning the output is the inner product of a family of basis functions with a signal. The parametrization and form of the basis functions determine the properties of the transforms. The number of basis functions for a complete picture (i.e. a result that contains enough information to reconstruct the original signal) is the same for all three cases.

For the standard Fourier transform the basis functions are simply the complex oscillations. It analyses the global frequency content in the image signal in terms of complex-valued coefficients associated with the frequencies. In addition, the frequency spectrum has an infinite set of bases. While this is good in terms of faithful reconstruction of the image, the complex valued parameters, and infinite spectrum have posed certain challenges and difficulties in understanding the local behaviours.

Discrete Cosine Transform (DCT) is closely related to the Discrete Fourier Transform (DFT) with some dissimilarity. The DCT is more efficient in concentrating energy into lower order coefficients than what the DFT does for image data. The DCT is purely real whereas the DFT is complex (magnitude and phase). However, DCT does not solve the issue of infinite support of spectrum.

The STFT adds a time dimension to the base function parameters by multiplying the infinitely long complex exponential with a window to localize it. One of the drawbacks of STFT is the fixed resolution property of the window.

The DWT does away with the constant bandwidth constraint and adapts the window size to the frequency and that happens in a specific scale invariant way such that it doesn't even need the complex modulation anymore. Wavelets allow both time and frequency analysis of signals simultaneously because the energy of wavelets is concentrated in time and still possesses periodic characteristics.

Neural networks have been used in watermarking methods for prediction of embedding regions or scaling factors mostly. End-to-end convolutional neural networks (CNNs) based watermarking schemes are very few compared to the existing spatial and frequency domain methods. Convolutional Neural Networks (CNNs) based watermarking methods are at par with existing frequency domain methods in the quality of the watermarked images but small perturbations made to the watermarked images affect the quality of the watermark extracted. In order to enhance the robustness of CNNs we provide them with wavelet coefficients of an image rather than the pixel values during training. In our study, we embed a grayscale watermark image onto an original grayscale image of the same size.

In section 2, we make a mention of studies that use wavelet transforms for copyright protection. In the same section, we also briefly cover the various popular existing methods for watermarking which use CNNs. We also discuss the multi-resolution property of DWT and how it is used in image processing in section III. In section IV, we highlight the main contribution of our work, and the proposed watermarking methodology.

Adversarial attacks are a part of CNN based methods for testing information security systems. From the perspective of watermarking we consider two such adversarial attacks in section 5. Experimental set up details in terms of the data set considered, the proposed CNN architecture, the assessment metrics used are elaborated in section 6. Quality assessment of watermarks extracted against different attacks is presented in section 7.

The contributions made in this study can be summarized as follows:

- A novel, robust and blind 3-level DWT based end-to-end CNN watermarking methodology.
- For purpose of copyright protection, we embed a grayscale logo image onto another grayscale image of the same size.
- Evaluation of the watermarking method against adversarial attacks in addition to signal processing and geometrical attacks on images. We demonstrate that our proposed system is robust against all forms of attacks even with the increased payload.

2. RELATED WORK

2.1 COPYRIGHT PROTECTION USING WAVELETS

Wavelet based watermarking methods provide good robustness to the watermark and thus are very useful for copyright protection of images and videos. In [1], the watermark image is embedded into the original image by first encrypting the watermark image and then applying DWT on the encrypted image. These wavelet coefficients are then combined with the wavelet coefficients of the original image. Devi and Singh [2] present a watermarking method for copyright protection of red-cyan anaglyph images using DWT, Hadamard transform and singular vector decomposition (SVD). Android based smartphones are in every nook and corner of the world. As a consequence, the number of images generated by these devices is huge. Hazem and Nour [3] propose a colour image watermarking scheme using DWT for copyright protection, this scheme is designed to be implemented for Android based smartphones. The robustness of DWT coefficients is even applied to videos. Preda and Dragos [4] and [5] provide a digital watermarking method for videos based on a multi-resolution wavelet decomposition. The watermark used is a binary image. This watermark is embedded in the wavelet coefficients of the LH, HL and HH sub-bands of the second wavelet decomposition level by quantization. This method is also useful since it is a blind strategy.

2.2 CNN BASED METHODS FOR IMAGE WATERMARKING

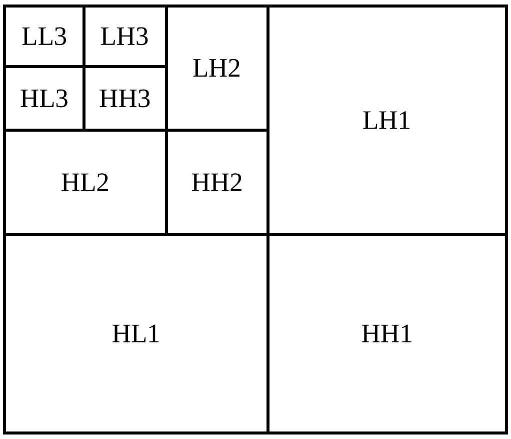
One of the first works that studied the possibilities of applying CNNs for digital watermarking is presented in [6]. Haribabu Kandi et al. [6] propose a non-blind digital image watermarking technique using auto-encoder functionality of CNN with codebook images. The drawback is the amount of information that the receiver needs in order to extract the binary watermark. In addition, in this work the CNN is used to generate the codebook images but not directly in the process of embedding or extracting the watermark. The comparison with existing spatial domain and frequency domain methods prove that CNNs can be used for image watermarking. Mun et al. [7] were the first to propose a deep learning based framework for watermarking. An autoencoder is used for embedding a binary watermark onto the encoded form of the cover image and the decoder converts this concatenated data to be perceptually similar to the cover image. A separate detector is trained to detect the watermark. The embedder and detector are not trained simultaneously. Zhu et al. [8] provided the first end-to-end deep learning framework for watermarking. A CNN based encoder is used to hide a binary secret message of length L in a colour image of dimension $H\times W$ where $L << H\times W$. A parameter-less noise layer is used to perform different attacks on the encoded image, meaning this layer does not undergo the training process of the network. The noise image is then given to a CNN based decoder to recover the secret message. A CNN based discriminator is trained to identify whether a given image is containing a secret message or not. Based on the work in [8], Ahmadi et al. [12] propose a watermarking framework called ReDMark. In addition to the framework adopted by Zhu et al., ReDMark introduce a differentiable approximation of JPEG in the attack layer. It also uses fixed DCT transform layers on the blocks of the cover image. Similar to [8], ReDMark embeds a binary watermark of size 4×4 onto a cover image of size 32×32 by dividing the image into blocks of size 8×8.

Liu et al. [9] provide a novel 2 stage deep learning framework for blind watermarking. The methods mentioned previously train the extracting unit to extract the secret message from attacked images only since the attack layer is made part of the training. Liu et al. claim that this is not a practical solution in reality since not all kinds of attacks performed on images can be differentiable. Hence, they propose a 2 stage method in which the extracting unit is first trained to extract the watermark from watermarked images which are not attacked. In the second stage, the watermarked images are attacked and the extracting unit is trained separately on these images. The extracting unit is trained and tested on conventional watermarking attacks like compression, noise addition, cropping and also software attacks like pencil sketch, crayon, starlight etc. Surprisingly they have not included the rotation attack.

Baluja [10] proposed a data hiding strategy based on encoder-decoder architecture using CNN. Three CNN are used for hiding an image within another image of the same size. The first CNN is called the Prep-Network: it prepares the secret image for hiding. Second network is the Hiding-network: it receives the secret image got from the prep-network and the cover image, the output of this autoencoder is called a container image, which is perceptually similar to the cover image. The last network is the Reveal-network: it takes as input the container image and returns the revealed image which is perceptually similar to the secret image. The work does not focus on watermarking specifically, hence conventional attacks studied for watermarking algorithms are not presented in their work.

3. DISCRETE WAVELET TRANSFORM

DWT is used in many image-processing applications. For example, the compression standard of JPEG 2000 uses wavelets for achieving good compression. It is preferred over Fourier transform due to its ability to represent both the temporal and frequency components of an image. The Fig.1 shows the different levels of wavelet decomposition of an image. LH^1, HL^1, HH^1 show the first level decomposition of the mid-frequency and high frequencies of an image. When a 2- level DWT is applied on an image, the LL^1 block is further split into LL^2, LH^2, HL^2, HH^2. In Fig.1 the LL^2 block is further broken down to third level frequency bands of LL^3, LH^3, HL^3, HH^3. The Fig.2 shows an original image of Lena and the 3-level DWT decomposition of the same. The overall size of the first level transformed image is same as the size of the spatial image. The individual sub-bands are half the size of the original image. In every further level, the size of the sub-bands is further halved.



H – High Frequency Bands

L – Low Frequency Bands

1, 2, 3 – Decomposition Levels

Fig.1. 3-level DWT of an image

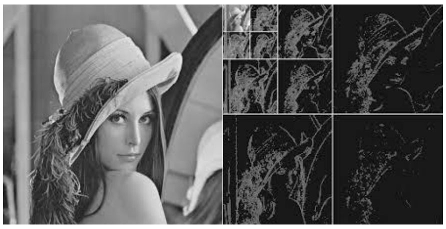


Fig.2. (a) Lena Image (b) 3-level DWT of Lena Image

The robustness of DWT in digital watermarking has also been established and studied extensively for the past two decades at least. Many methods combine wavelet transforms with other methods like SVD, DCT etc. It is not possible to mention all the works here but we mention a few of the popular recent studies for the interested reader. [13], [14], [15], [16], [17].

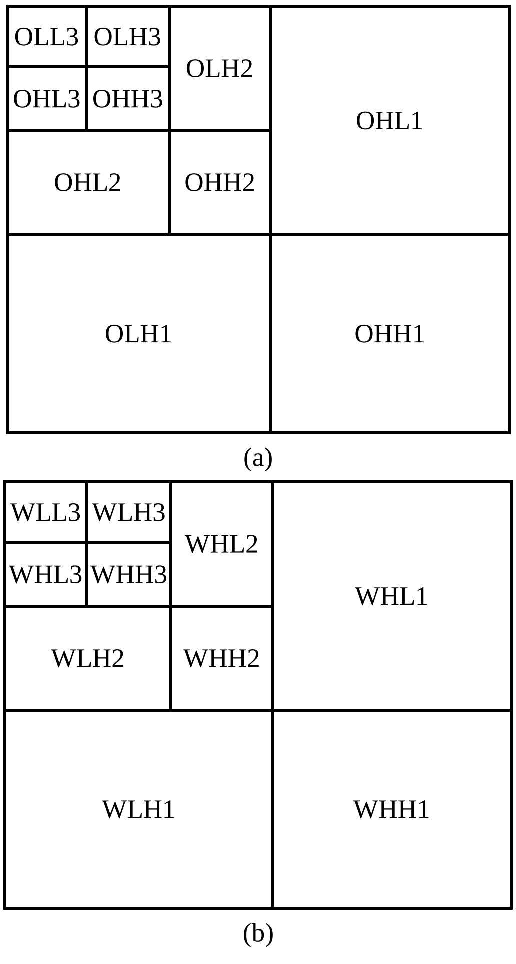
4. PROPOSED METHOD

The proposed method is split into 2 stages, we provide the details of the methodology stage wise.

4.1 STAGE I

We use the 3-level DWT transform on the set of cover images and the watermark image. The different subbands got after applying DWT on the original image and the watermark image are shown in different colours in Fig.3 as OLL_3, OHL_3,..., OHH_1 where O is used to denote the sub-bands of the original image. Similarly we have WLL3, WHL_3,..., WHH_1 where W is used to denote the sub-bands of the watermark image.

All the wavelet bands of the original image and the watermark image are given as input to the embedding network Embed I as shown in Fig.4. The output of Embed I is denoted as the WMKED Bands I. After 300 epochs, we expect the WMKED Bands I to be as close as possible to the different wavelet bands of the original image. Embed I is trained with the loss function provided in Eq.(1), MSE refers to Mean Square Error. In Eq.(1) $input_1$ denotes the sub-bands of the original image and $output_1$ denotes the output of Embed I. In every iteration the batch of output of Embed I are provided to the extracting network Extract I and this is trained to extract the sub-bands of the watermark image, denoted as EXT WMK Bands I. The extractor network is trained on a combined loss function which is provided in Eq.(2). The $input_2$ refers to the sub-bands of the watermark image provided as input to Embed I and $output_2$ refers to the output of Extract I. We use the combined loss in order to maintain imperceptibility of the watermarked image as well as the robustness of the watermark.



(a)

(b)

Fig.3. (a) 3- Level DWT of Original Image (b) 3-Level DWT of Watermark Image

$$Loss(input_1, output_1) = MSE(input_1, output_1) \qquad (1)$$

$$CombLoss(input_1, input_2, output_1, output_2) = MSE(input_1, output_1) + MSE(input_2, output_2) \qquad (2)$$

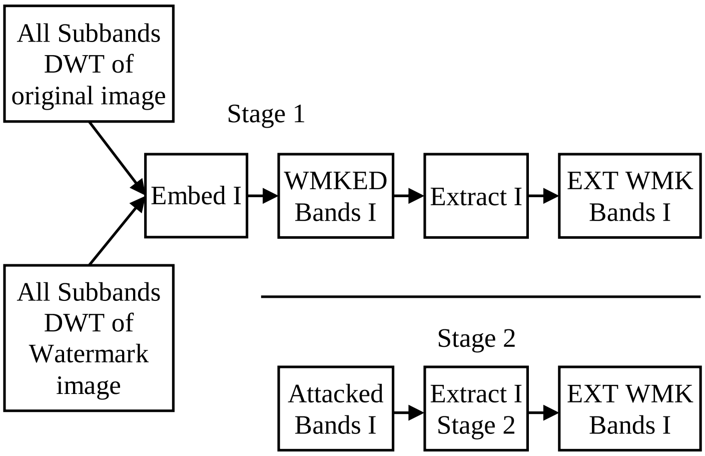


Fig.4. All DWT subbands of (a) original and (b) watermark image

The second level wavelet bands of the original image and the watermark image are given as input to the embedding network EMBED II. The Fig.5 depicts this step. In each iteration, the output of EMBED II is provided to the extracting network EXTRACT II. The respective outputs of these networks is expected to be close to the inputs provided to EMBED II. As explained in detail in step 2 for all the sub-bands, in this case we expect the same for the second and third sub-bands.

Similar to the above two steps, third level of wavelet bands of the original image and the watermark image are provided as input to the embedding network Embed III. The output of Embed III is provided as input to the extracting network Extract III. This is shown in Fig.6. It is important to note that the six networks explained above are trained simultaneously in pairs.

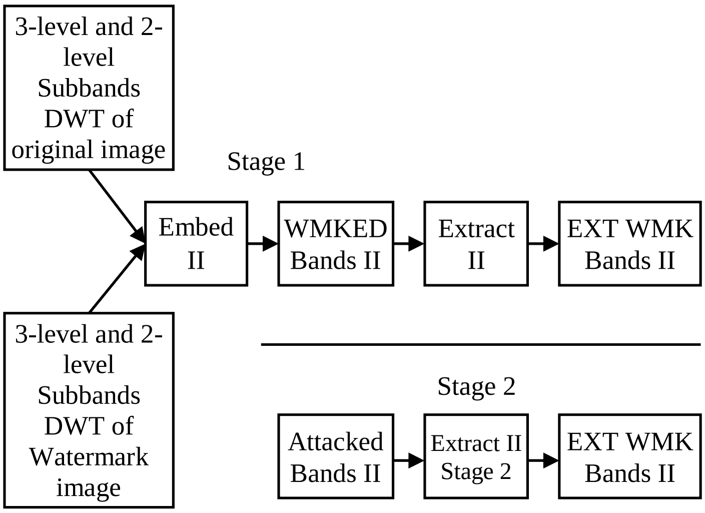


Fig.5. 3-Level and 2-Level DWT subbands of (a) original and (b) watermark image

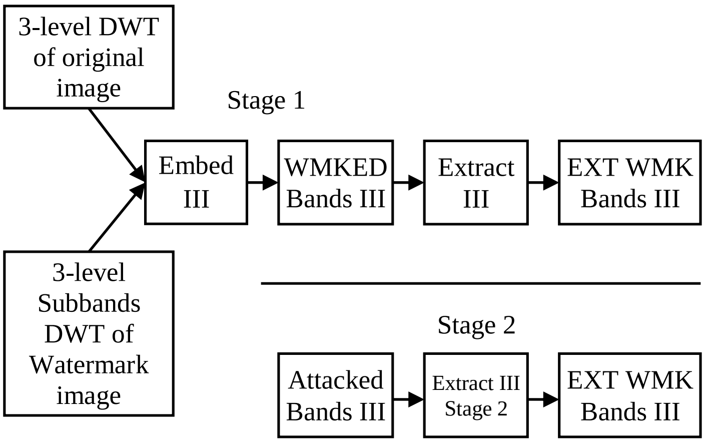


Fig.6. 3-Level DWT subbands of (a) original and (b) watermark image

The cover image/watermarked image, that contains the watermark is obtained by combining the outputs of Embed I, Embed II and Embed III as shown in Fig.7. In essence, we take the third level watermarked sub-bands from Embed III, the second level watermarked sub-bands from Embed II and the first level watermarked sub-bands from Embed I. We combine these sub-bands and apply the 3-level IDWT to get the watermarked image.

Similarly, the extracted watermark is obtained by combining the outputs of the three extracting networks Extract I, Extract II and Extract III and applying 3-level IDWT as shown in Fig.8.

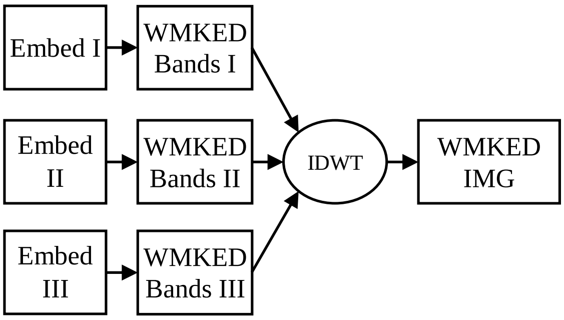


Fig.7. Combination of Watermarked image using Embed I, II, III

4.2 STAGE II

In order to test robustness of the extractor network we provide an input of watermarked images that are attacked by either a signal processing operation or geometric operations. It is observed that when such images are provided to the extractor network Extract I, the quality of the extracted watermark is not good enough. In order to improve this functionality of the extractor networks, we create extractor networks whose parameters are initialized to the final value of the parameters of the corresponding network. For example, for Extract I a new extracting network Extract I Stage 2 is created. This is shown in Figures 4, 5 and 6, where a line separates the activity in the 2 stages.

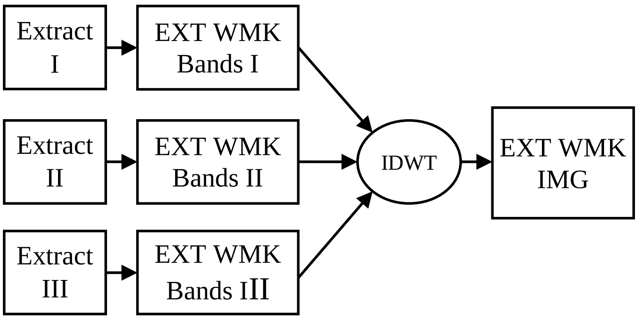


Fig.8. Combination of Watermarked image using Extract I, II, III

4.3 SINGLE-LEVEL DWT METHOD

To highlight the significance of the proposed methodology we compare the 3-level DWT's performance with single-level DWT. In the case of a single level DWT there are no multiple bands, hence the training and testing of single-level DWT is similar to the method shown in Fig.4. In this case, the watermarked image and the extracted watermark are got by applying IDWT to the output of Embed I and Extract I. While single-level DWT is also a blind method and carries the same payload information as the 3-level DWT, The 3-level DWT is significantly more robust compared to single level DWT.

In this study the robustness of the watermarking methods is tested against signal processing attacks like compression, Gaussian blur, median filtering and salt and pepper noise. We also study the robustness against geometrical attacks like rotation and cropping. The extent of modifying the watermarked image must be such that the modified image is still in a good visible condition. In this study we consider two adversarial attacks which keep the perceptual quality of the watermarked image but attempt to either destroy or remove the watermark information in an invasive manner.

5. ADVERSARIAL ATTACKS

- *Convolutional Autoencoder Attack*: Auto encoder is an artificial neural network used for dimensionality reduction of data. There are two parts to an autoencoder i.e. the encoder and the decoder. The encoder reduces the input to a state space with fewer dimensions and the decoder reconstructs the input from that representation. It learns to ignore the latent noise in the higher dimensions of the data. In [19], the authors propose a CNN based attack on watermarked images. A CNN autoencoder is used to create images that perceptually match the watermarked images. This recreated image, (output of the CNN) is visually very close to the watermarked image. The modifications in the CNN auto encoder do affect the quality of the extracted watermarks.
- *Double Watermarking Adversarial Attack*: Watermarking algorithms are available publicly. Invisible watermarking methods do not change the perceptual quality of an image. One cannot say with naked eyes if an image is watermarked or not in the case of invisible watermarking. This provides a situation to an adversary who can also hide a watermark within already watermarked images. The addition of a second watermark makes it difficult to extract/detect the original watermark present. Both these attacks are also studied in this work. The results for the same are presented in section 7.

6. EXPERIMENTAL SETUP

In order to prove the robustness of the multi-level DWT coefficients we compare the performance of a single-level DWT with a 3-level DWT. The 3-level DWT methodology has already been described in section IV. In single level DWT method, only the first level wavelet transform coefficients are used. Rest of the embedding and extraction procedure of the single level DWT is the same as 3-level DWT.

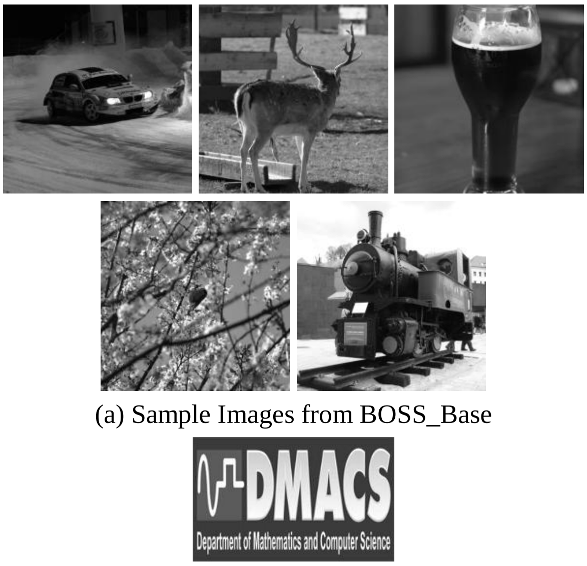
6.1 DATASET

We have used the BOSSBase 1.01 data set [20]. It consists of 10000 grayscale images for training and 2000 images for testing, each of size 512×512. We have used 2000 images in our study for training and 400 images for testing. Five images from this data set and the watermark logo used are shown in Fig.9. For our training and testing purposes, we have resized the images of the data set to 64×64. The grayscale watermark image is also of the same size (64×64). The extractor networks are trained on a dataset of attacked images. This dataset of attacked images contains around 10000 attacked watermarked images.

6.2 CNN FRAMEWORK

The framework consists of three pairs of CNNs in Stage I as explained in the proposed methodology.

- Embed CNN I and Extract CNN I
- Embed CNN II and Extract CNN II
- Embed CNN III and Extract CNN III



(a) Sample Images from BOSS_Base

(b) Watermark Image

Fig.9. Dataset Image and image for watermarking

The architectures of the Embed and Extract networks is shown in Table.1. The adaptive moment estimation (Adam) optimizer is used with a learning rate of 0.001. Adam is used since it is proved to have better loss reduction. All the networks are trained for 300 epochs. Pytorch framework is used for the entire deep learning frameworks. Python libraries scikit and OpenCV are used for the image processing tasks. The network was trained on TESLA K40 GPU.

6.3 WATERMARKING QUALITY ASSESSMENT METRICS

In order to test the quality of the watermarked image we compare and measure its perceptual similarity with the original non-watermarked image. This similarity is measured using Peak Signal to Noise Ratio (PSNR) measure.

- *Peak Signal to Noise Ratio (PSNR)*: Given two images I and J, where J is the noisy version of image I, the PSNR value provides the extent to which noise is present in the image J. A low PSNR value indicates more noise in the image. PSNR is usually expressed in decibel units. PSNR is calculated as follows:

$$PSNR=10\log_{10}(MAX_I/MSE) \tag{1}$$

where MSE is the Mean Square Error between the intensity values of the pixels in the two images I and J and MAX_I denotes the maximum intensity value of a pixel in the image I. In order to test the quality of the extracted watermark after the watermarked image has been manipulated by different kind of attacks, we use Structural Similarity Index Measure (SSIM). This measure is a better indicator of the quality of the watermark extracted since PSNR values of the extracted watermark will be sensitive to any noise in the image.

Structural Similarity Index Measure (*SSIM*): Unlike PSNR and MSE which are based on absolute errors, this metric is based on similarity between the structural information in two images I and J. SSIM is evaluated on various windows of an image. Given 2 windows x and y of common size ($N*N$) SSIM is calculated as follows [21]:

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

with μ_x the average of x, μ_y the average of y, σ^2 is the variance of x, σ^2 is the variance of y, σ_{xy} the co-variance of x and y, $c_1 = (k_1L)^2$,$c_2 = (k_2L)^2$ are two variables to stabilize the division with the weak denominator, L is the dynamic range of the pixel values, $k_1 = 0.01$ and $k_2 = 0.03$ by default.

- *Normalized Cross Correlation (NCC)*: This is used for template matching between two images I and J by the following formula:

$$NCC\left(I,J\right)=\frac{1}{n}\sum_{x,y}\frac{1}{\sigma_x\sigma_y}I\left(x,y\right)J\left(x,y\right) \tag{3}$$

where n is the number of pixels in the template.

Table.1. Architecture of the Embed and Extract networks

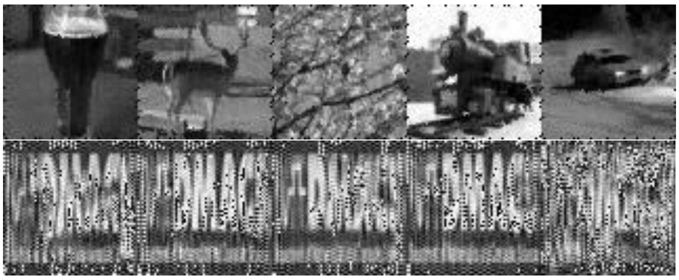
Network	Input	Layer (filter,stride, depth,activation)	Output
Embed	Original DWT Coefficients (64*64*1) + Watermark DWT Coefficients (64*64*1)	Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 1)	Watermarked DWT Coefficients (64*64*1)
Extract	Watermarked DWT Coefficients (64*64*1)	Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 32, ReLU) BatchNorm2d (32) Conv2d (3*3, 1*1, 1)	Extracted DWT Watermark Coefficients (64*64*1)

7. RESULTS

The quality of the watermark extracted by a single-level DWT and multi-level DWT after various attacks is presented here. The extracted watermarks presented here are extracted by the extractor network trained on attacked images, i.e., EXTRACT II. The results are presented for every attack separately. In every figure, the top row shows the image after the attack on the watermarked image, the middle row shows the corresponding watermark extracted using single-level DWT and the final row shows the watermark extracted using 3-level DWT.

7.1 ROTATION

Watermarked images are rotated in a range of (60°, 60°) and given as input as part of the package of attacked images to Extract II network. After the Extract II network is trained, the test set images are rotated at an angle of 60° and again rotated by 60° to get back the original image size after doing interpolation and cropping. The results for the extracted watermarks is shown in Fig.10.



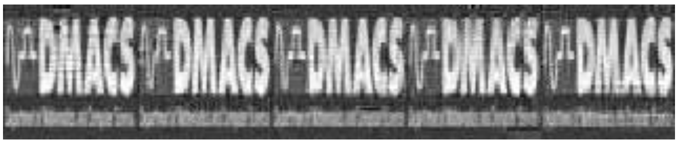


Fig.10. Extracted watermarks using 1-level and 3-level DWT against rotation attack

7.2 CROPPING

Cropping is again a geometric attack on the images like rotation. In cropping, the chances of losing data are higher. Watermarked images are cropped by (25%) and given as input as part of the package of attacked images to train Extract II network. After the Extract II network is trained, the test set images are cropped by 15%. Watermark extraction is then performed on these modified images. The results for the extracted watermarks is shown in Fig.11.

7.3 MEDIAN FILTER

Median filters are usually applied to images in order to remove noise from the images. This is a non-linear digital filtering technique. From our perspective this noise removal can lead to loss of some information from the image, this could even be loss of watermark information in the image. Watermarked images are filtered with different filter sizes of (3, 3) and (5, 5) and given as input as part of the package of attacked images to train the Extract II network. After the Extract II network is trained, the test set images are filtered using a (3, 3) filter. Watermark extraction is then performed on these modified images. The results for the extracted watermarks is shown in Fig.12.

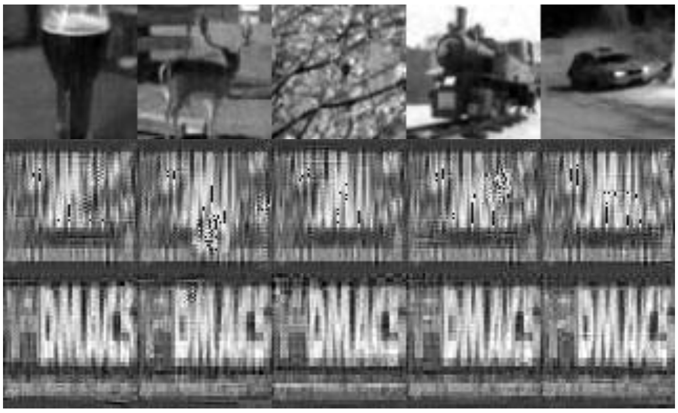
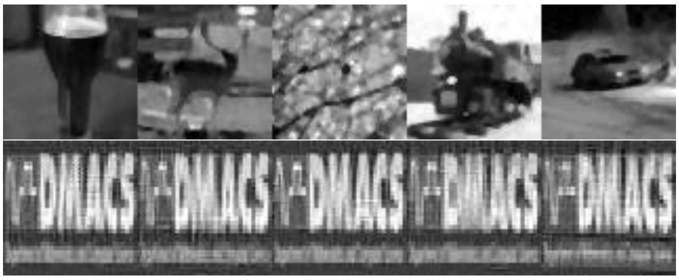


Fig.11. Extracted watermarks using 1-level and 3-level DWT against cropping attack



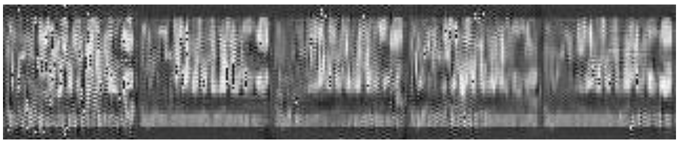


Fig.12. Extracted watermarks using 1-level and 3-level DWT against median filtering attack.

7.4 SALT AND PEPPER NOISE

Median filters are usually applied to images which have salt and pepper noise in them. Salt and pepper noise results due to sudden disturbances in the image signal. With the watermarked data existing in public domain, it is highly likely that the watermarked data undergoes this effect. There by we add Sand P noise of different intensities to the watermarked images and these images are then given as input as part of the package of attacked images to train the Extract II network. After the Extract II network is trained, a SandP noise with mean 0 and variance 0.05 is added to all the test images. Since the perceptual quality of the image needs to be maintained well, only a small amount of noise is added. Watermark extraction is performed on these modified images. The results for the extracted watermarks is shown in Fig.13.

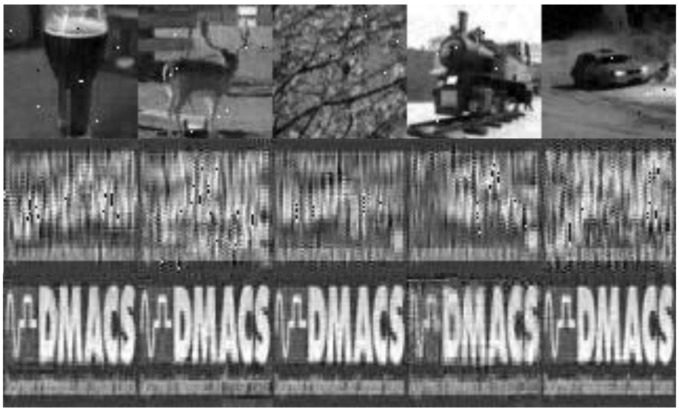


Fig.13. Extracted watermarks using 1-level and 3-level DWT against salt and pepper noise attack.

7.5 GAUSSIAN BLUR

Gaussian blurring of images results in blurring of the images by a Gaussian function. It is also used to hide details of an image by smoothing the image content. When Gaussian blur is applied to images, it can result in loss of data that is present in edges in the image. This is captured by high frequencies in an image. Gaussian filters with different mean and variance are applied on the watermarked images and this is provided as input as part of the package of attacked images to Extract II network. For testing purposes we apply Gaussian blur filter of size (3,3) with mean 0 and standard deviation 0.8. Watermark extraction is then performed on these modified images. The results for the extracted watermarks is shown in Fig.14.

7.6 JPEG COMPRESSION

JPEG compression is a very common image processing operation done on images of high quality. For every good resolution image available on the internet there is always a jpeg compressed version available. Watermarked images undergo different levels of JPEG compression. These compressed im- ages are then added as part of the input of attacked images used to train Extract II network. For testing purposes we apply 20% JPEG compression on the test images. Watermark extraction is then performed on these modified images. The results of the extracted watermarks is shown in Fig.15.

7.7 CONVOLUTIONAL AUTOENCODER ATTACK

The details of this attack have already been mentioned in section 5. Test images of the dataset are watermarked by the Embed network. The watermarked images are provided to the autoencoder. The extract network extracts the watermark from the reconstructed images of the autoencoder. We do not train the Extract II network on these images. The results of the extracted watermarks is shown in Fig.16.

7.8 CONVOLUTIONAL AUTOENCODER ATTACK

The details of this attack have already been mentioned in section V. Images watermarked with the logo image are provided as input to a network trained to embed Lena image shown in Fig.2. From these double watermarked images, we use the Extract II network to find the original (logo) watermark. In Fig.17 we show the five sample images that are double watermarked and the logo image that is extracted from them.

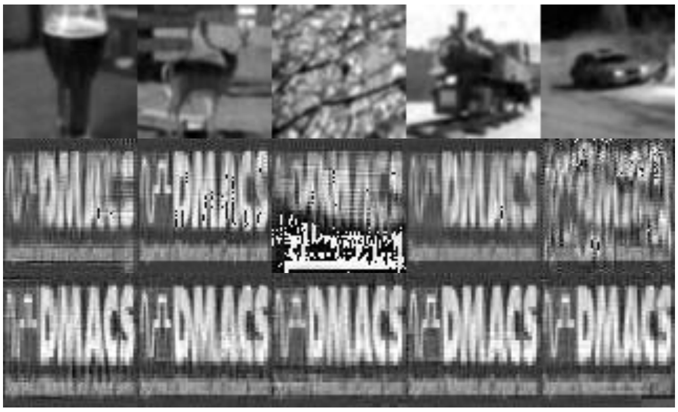


Fig.14. Extracted watermarks using 1-level and 3-level DWT against Gaussian blur attack

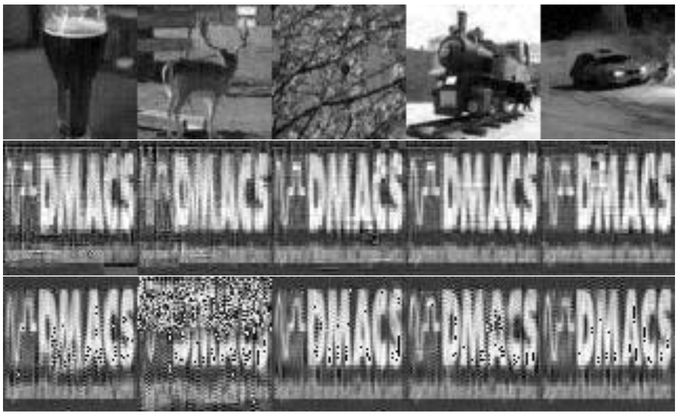


Fig.15. Extracted watermarks using 1-level and 3-level DWT against JPEG compression attack

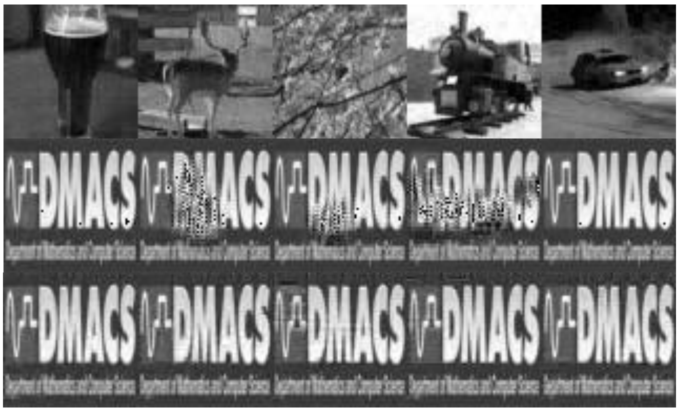


Fig.16. Extracted watermarks using 1-level and 3-level DWT against CAE attack

Table.2. Quality of watermarked images

Method	SSIM	PSNR	NCC
1-level DWT	0.9166	30.85	0.9964
3-level DWT	0.9737	31.89	0.9974

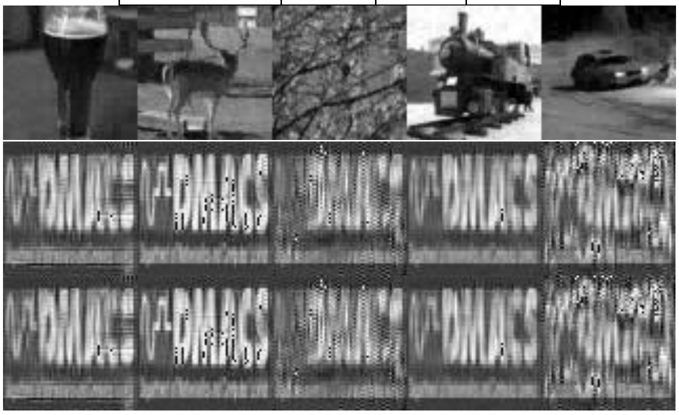


Fig.17. Extracted watermarks using 1-level and 3-level DWT against double watermark attack.

7.9 NUMERICAL RESULTS

Imperceptibility of the watermark is an important feature that must be maintained by watermarking methods. In our study we show the quality of the watermarked images using the 2 methods of single-level DWT and multi-level DWT on 400 images taken from the BOSSBase test dataset in In Table.2. In order to test the methods of single-level DWT and 3-level DWT we test the STAGE 2 extractor networks on the same test set mentioned here. The values of PSNR, SSIM and NCC presented in Table.3 are a simple average of the values obtained for the 400 images. The first row denotes the values of the watermark extracted from unattacked watermarked images.

Table.3. Quality of watermarks extracted

Attack	Method	SSIM	PSNR	NCC
Unattacked	1-level DWT 3-level DWT	0.9196 0.9737	23.19 29.78	0.9927 0.9984
Rotation (60)	1-level DWT 3-level DWT	0.6205 0.8687	16.82 19.02	0.9702 0.9793

Cropping (15)	1-level DWT	0.1422	10.84	0.8698
	3-level DWT	0.6001	16.82	0.9664
Median Filter (3,3)	1-level DWT	0.4291	13.48	0.9355
	3-level DWT	0.7460	18.91	0.9807
Salt and Pepper Noise Mean =0, sd = 0.05	1-level DWT	0.7334	17.57	0.9771
	3-level DWT	0.9193	24.50	0.9944
Gaussian Blur (3,3) Mean=0, sd = 0.8	1-level DWT	0.5008	14.47	0.9513
	3-level DWT	0.8022	20.12	0.9853
JPEG Compression (20 %)	1-level DWT	0.6834	19.51	0.9830
	3-level DWT	0.8171	20.55	0.9863
Convolutional Autoencoder Attack	1-level DWT	0.8160	20.22	0.9700
	3-level DWT	0.9262	24.95	0.9950
Double Watermarking Attack	1-level DWT	0.6326	15.32	0.9530
	3-level DWT	0.7471	18.67	0.9802

7.10 ANALYSIS

At the completion of the study undertaken in this work, we sum up our findings as follows.

- On comparison of the extracted watermarks by the single-level DWT and multi-level DWT it is clear that the multi-level DWT is more robust to attacks. It would be tempting to embed the watermark in the deeper levels of the DWT transform unlike the method adopted in this paper. While that technique will provide robustness in comparison with the technique proposed in this work, what makes this work unique is the blind methodology. If only the deeper levels of the original image and the watermark are used for watermarking one would need to store additional information from the sub- bands to reconstruct the watermarked image and extract the watermark from it. Blind methods are more practical in real life situations.
- Many watermarking methods embed a watermark which is a binary image or a binary sequence, while in this work we choose to embed a grayscale image which contains more pictorial information. From the perspective of copyright protection, we find such logos more useful. There by we also prove that this methodology can work for data hiding purposes.
- Given 2000 original images with no watermark information embedded in them to the extractor networks result in more or less junk information. The extracted watermarks have average values of SSIM, PSNR, and NCC as 0.2685, 10.38, and 0.9056 respectively. We undertake this study to demonstrate that the extractor network will not extract meaningful watermarks information from non-watermarked images.

8. CONCLUSION

Copyright protection of multimedia assets like images is an important area of research in today's scenario of the internet where so many image sharing platforms exist. In this study, we present a novel way of combining the robustness of multi-resolution wavelet transform with the learning ability of CNNs to provide a blind method for watermark extraction. In the proposed method, we also show that it is possible to hide a full size grayscale image into a cover image of the same size. This combination makes the proposed algorithm practical and useful. It is also shown in this work that the method is robust against adversarial attacks not studied largely in earlier watermarking studies.

REFERENCES

[1] Sarita P Ambadekar, Jayshree Jain and Jayshree Khanapuri, "Digital Image Watermarking through Encryption and DWT for Copyright Protection", *Recent Trends in Signal and Image Processing*, pp.187-195, 2019.

[2] Hidangmayum Saxena Devi and Khumanthem Manglem Singh, "Red- Cyan Anaglyph Image Watermarking using DWT, Hadamard Transform and Singular Value Decomposition for Copyright Protection", *Journal of Information Security and Applications*, Vol. 50, No. 1, pp. 1-17, 2020.

[3] Hazem Al-Otum and Nour Emad Al-Shalabi, "Copyright Protection of Color Images for Android-Based Smartphones using Watermarking with Quick-Response Code", *Multimedia Tools and Applications*, Vol. 77, pp. 1-24, 2018.

[4] Radu O Preda and Dragos N Vizireanu, "A Robust Digital Watermarking Scheme for Video Copyright Protection in the Wavelet Domain", *Measurement*, Vol. 43, pp. 1720-1726, 2010.

[5] Radu O Preda and Dragos N Vizireanu, "Robust Wavelet-Based Video Watermarking Scheme for Copyright Protection using the Human Visual System", *Journal of Electronic Imaging*, Vol. 20, No. 1, pp. 1-19, 2011.

[6] Haribabu Kandi, Deepak Mishra and Subrahmanyam R.K. Sai Gorthi, "Exploring the Learning Capabilities of Convolutional Neural Networks for Robust Image Watermarking", *Computers and Security*, Vol. 65, pp. 247-268, 2017.

[7] Seung-Min Mun, Seung-Hun, Jang Nam, Kim Haneol and Heung-Kyu Lee Dongkyu," Finding Robust Domain from Attacks: A Learning Framework for Blind Watermarking", *Proceedings of 27th ACM International Conference on Multimedia*, pp. 191-202, 2019.

[8] Jiren Zhu, Russell Kaplan, Justin Johnson and Li Fei-Fei, "Hidden: Hiding data with Deep Networks". *Proceedings of European Conference on Computer Vision*, pp. 657-672, 2018.

[9] Yang Liu, Mengxi Guo, Jian Zhang, Yuesheng Zh and Xiaodong Xie, "A Novel Two-Stage Separable Deep Learning Framework for Practical Blind Watermarking", *Proceedings of ACM International Conference on Multimedia*, pp. 1509-1517, 2019.

[10] Shumeet Baluja,"Hiding Images within Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 42, No. 7, pp. 1685-1697, 2019.

[11] Bingyang Wen and Sergul Aydore, "Romark: A Robust Watermarking System using Adversarial Training", *Proceedings of ACM International Workshop on Machine Learning*, pp. 1-5, 2019.

[12] Mahdi Ahmadi, Alireza Norouzi, Nader Karimi and Shadrokh Samavi, "ReDMark: Framework for Residual Diffusion Watermarking based on Deep Networks", *Expert Systems with Applications*, Vol. 146, pp. 1131-57, 2020.

[13] Musrrat Ali and Chang Wook Ahn, "An Optimized Watermarking Technique based on Self-Adaptive DE in DWT-SVD Transform Domain", *Signal Processing*, Vol. 94, No. 2, pp. 545-556, 2014.

[14] Tanya Koohpayeh Araghi, Azizah Abd Manaf and Sagheb Kohpayeh Araghi. "A Secure Blind Discrete Wavelet Transform based Watermarking Scheme using Two-Level Singular Value Decomposition", *Expert Systems with Applications*, Vol. 112, pp. 208-228,2018.

[15] Reem A Alotaibi and Lamiaa A Elrefaei, "Text-Image Watermarking based on Integer Wavelet Transform (IWT) and Discrete Cosine Transform (DCT)", *Applied Computing and Informatics*, Vol. 15, No. 1, pp. 191-202, 2018.

[16] Ladan Salimi, Amir Haghighi and Abdolhossein Fathi, "A novel Watermarking Method based on Differential Evolutionary Algorithm and Wavelet Transform", *Multimedia Tools and Applications*, Vol. 81, pp. 1-18, 2020.

[17] Mohammad Hassan Vali, Ali Aghagolzadeh and Yasser Baleghi. "Optimized Watermarking Technique using Self-Adaptive Differential Evolution based on Redundant Discrete Wavelet Transform and Singular Value Decomposition", *Expert Systems with Applications*, Vol. 114, pp. 296-312, 2018.

[18] Falgun N. Thakkar and Vinay Kumar Srivastava, "A Blind Medical Image Watermarking: DWT-SVD based Robust and Secure Approach for Telemedicine Applications", *Multimedia Tools and Applications*, Vol. 76, pp. 3669-3697, 2017.

[19] Sai Shyam Sharma and V. Chandrasekaran, "A Robust Hybrid Digital Watermarking Technique against a Powerful CNN-based Adversarial attack", *Multimedia Tools and Applications*, Vol. 84, pp. 1-22, 2020.

[20] Break our Steganographic System, Avaialble at http://agents.fel.cvut.cz/boss/index.php?mode=VIEW%20tmpl=materials, Accessed at 2020.

[21] Zhou Wang, Eero P. Simoncelli and Alan C. Bovik, "Multiscale Structural Similarity for Image Quality Assessment", *Proceedings of 37th Asilomar Conference on Signals, Systems and Computers*, pp. 1398-1402, 2003.