Using Deep learning for image watermarking attack

Makram Hatoum, Jean-François Couchot, Raphael Couturier, Rony Darazi

Using Deep Learning for Image Watermarking Attack

Makram W. Hatoum[a,*], Jean-François Couchot[a], Raphaël Couturier[a], Rony Darazi[b]

[a] *FEMTO-ST Institute, University of Bourgogne Franche-Comté, UMR 6174 CNRS, France. www.univ-fcomte.fr*
[b] *TICKET Lab, Antonine University, Hadat-Baabda, Lebanon. www.ua.edu.lb*

Abstract

Digital image watermarking has justified its suitability for copyright protection and copy control of digital images. In the past years, various watermarking schemes were proposed to enhance the fidelity and the robustness of watermarked images against different types of attacks such as additive noise, filtering, and geometric attacks. It is highly important to guarantee a sufficient level of robustness of watermarked images against such type of attacks. Recently, Deep learning and neural networks achieved noticeable development and improvement, especially in image processing, segmentation, and classification. Therefore, in this paper, we studied the effect of a Fully Convolutional Neural Network (FCNN), as a denoising attack, on watermarked images. This deep architecture improves the training process and denoising performance, through which the encoder-decoder remove the noise while preserving the detailed structure of the image. FCNNDA outperforms the other types of attacks because it destroys the watermarks while preserving a good quality of the attacked images. Spread Transform Dither Modulation (STDM) and Spread Spectrum (SS) are used as watermarking schemes to embed the watermarks in the images using several scenarios. This evaluation shows that such type of denoising attack preserves the image quality while breaking the robustness of all evaluated watermarked schemes. It could also be considered a deleterious attack.

Keywords: Image denoising, Deep learning, Digital watermarking,

*Corresponding author at: FEMTO-ST Institute, University of Bourgogne Franche-Comté, UMR 6174 CNRS, France. www.univ-fcomte.fr

Email address: makram.hatoum@univ-fcomte.fr (Makram W. Hatoum)

1. Introduction

Protection of digital contents was and still one of the most important topics in scientific research. With the progression of internet technologies, unauthorized users illegaly duplicate, authenticate, and distribute digital contents. Therefore, various watermarking methods have been studied for a wide range of applications, such as broadcast monitoring, copyright protection, content authentication, and copy control [1]. The embedded watermark could be a single bit or multi-bit generated from a pseudo-random sequence, obtained from a pseudo-random number generator. Also, this watermark could be a binary image or a gray-scale image.

The watermarking schemes are generally classified as additive class known by Spread Spectrum (SS) [2, 3, 4], and substitutive class known by Quantization Index Modulation (QIM) [5]. Spread Transform Dither Modulation (STDM) [6] is a special case of QIM, which has distinguished robustness against requantization and random noise attacks. STDM combines the robustness of SS and the effectiveness of QIM.

The watermarking algorithms are characterized by several properties such as payload, robustness, and fidelity. Therefore, the transform domains such as SVD, DCT, and DWT are usually used. The DCT domain is more robust than the spatial domain. Especially against simple image processing operations like brightness, blurring, and low pass filtering [7]. Also, the DWT is a very attractive transform which makes the watermarked images much more robust. The DWT domain composes the image in different levels of resolution and processed from high resolution to low resolution [8]. Hiding the watermark with more energy in an image will enhance the level of robustness. Furthermore, the Singular Value Decomposition (SVD) has been widely used for digital image watermarking. The SVD preserves the visual perception of the cover image and good robustness against most types of attacks [9, 10].

In the latest years, Artificial Intelligence (AI) and Deep Learning fields have exploded as computers and servers get closer to delivering human-level capabilities. Nowadays, companies around the world are looking to use their big data sets as a training ground to develop programs that can interact with the world in more natural ways, and extract from it useful information that has never been done before. Deep Learning and Neural Networks currently

provide a perfect solution to many problems in speech recognition, image recognition, and natural language processing [11, 12, 13, 14]. Moreover, the Convolutional Neural Network (CNN), which is a type of artificial neural network, has been widely used for image processing, segmentation, classification, and other auto-correlated data[15, 16, 17, 18]. Image denoising and super-resolution are topics of great interest in image processing that can lead to an improvement in image quality. Lately, the denoising accuracy is performed by deep neural networks by creating a mapping between the clean and noisy images [19, 20, 21, 22]. The successful results of CNN for image denoising are assigned to its large modeling capacity and enormous advances in network training and design. CNN with deep architecture effectively increases the flexibility for exploiting image characteristics and improving the training process and denoising performance. The advances are achieved with the learning methods for training CNN, including batch normalization and Rectifier Linear Unit (ReLU). This type of denoising could be harmful to watermarked images since the embedded watermark is like a sequence of noise embedded in the images.
Image filtering and denoising are a dangerous type of attacks in digital watermarking since it recovers the original value for each pixel of the image [23, 24, 25]. In this paper, we studied the effect of a Fully Convolutional Neural Network (FCNN) against watermarked images. We evaluated such type of denoising against SS and STDM watermarking schemes, and whether it could be used as a new type of attack in digital watermarking.
The remainder of this paper is organized as follows. Related work of watermarking attacks is presented in Section 2. Section 3 recalls some backgrounds on STDM and SS watermarking schemes. Section 4 briefly presents the architecture of the Convolutional Neural Network. The Fully Convolutional Neural Network is presented in Section 5. The evaluation of the proposed attack is presented in Section 6. Finally, in Section 7 we give our conclusion and future work.

2. Related Works

The watermarked images could be affected by different types of attacks, such as additive noise, lossy compression, geometric distortions, and image filtering attacks [23, 24]. The most common types of noise attack are salt&pepper noise and additive Gaussian noise attacks. Salt&pepper noise alters the pixel value to 0 or 255 (black and white) for an 8-bit gray-scale

image and the additive Gaussian noise reduces the visual quality of the image.

Median, Wiener, Average, and Gaussian filters are part of image filtering attacks that could destroy the watermark embedded in the watermarked images. The median filter is a non-linear digital filtering technique, which preserves the edges in the image while removing noise. Wiener filter is usually used for removal of blur in images. The average filter reduces the amount of intensity variation between pixels; each pixel value is replaced with the mean value of its neighbors, including itself. The Gaussian filter usually used to blur the image and to reduce contrast and noise.

Geometric attacks are geometric distortions to an image which include operations such as scaling, rotation, cropping, and translation [26]. They are classified basically into local and global geometric attacks. Local geometric attacks affect portions of an image using such as the cropping attack, and the global geometric attacks affect all the pixels of an image using such as the rotation and the scaling attacks.

Several methods were proposed to improve the robustness against geometric attacks. Enping Li et al. [27] presented a blind image watermarking scheme using a wavelet tree quantization to enhance the robustness against geometric attacks such as rotation, scaling, and cropping. Liu et al. [28] introduced a robust multi-scale full-band image watermarking based on the Singular Value Decomposition (SVD) and the Distributed Discrete Wavelet Transform (DDWT). This method has good robustness against cropping and rotation attacks. Li [29] proposed a robust image watermarking scheme based on a computer-generated hologram against geometric attacks, including translation, rotation, cropping, flipping, and scaling attacks. He et al. [30] proposed an image watermarking algorithm based on histogram modification resistant to geometrical attacks, including rotation, cropping, scaling, and translation attacks. Fazli and Moeini [31] presented a robust image watermarking method based on DCT, SVD, and DWT for the correction of geometric attacks. This method enhances the robustness against cropping, translation, and rotation attacks.

JPEG compression and Fixed Gain Attack (FGA) are also a type of attacks that could destroy the embedded watermark in the watermarked image. FGA attack changed the brightness and darkness of watermarked images based on a gain factor. Li et al. [32] improved the STDM watermarking scheme using a perceptual model to enhance the robustness against JPEG compression. Lin et al. [33] improved an image watermarking technique against JPEG com-

pression. The watermark is embedded in the low-frequency coefficients after applying the DCT frequency transform on the original image. Li and Cox [34] improved the robustness of STDM against amplitude scaling and JPEG compression using a perceptual model based on Watson's model. Hatoum et al. [35] proposed a blind image watermarking using Normalized STDM robust against FGA attack, AWGN attack, and JPEG compression.
Recently, CNN was exploited for modeling images priors for denoising. Zhang et al. [36] proposed an architecture called Deep network CNN (DnCNN) for image denoising. This deep network is composed of layers with different convolutional blocks using a convolution kernel size of 3×3. The first layer has Convolution + Relu, the intermediate layers have Convolution + BatchNorm + Relu, and the last layer has only Convolution. Using this architecture, the networks of 17 and 20 layers that are trained for blind denoising with specific noise level for additive Gaussian denoising, slightly outperformed the WNNM [37] and BM3D [38] denoising methods.
Another architecture called FFDNet was introduced by Zhang et al. [39] that handles a wide range of noise levels. FFDNet has a similar CNN architecture as of DnCNN, which is developed for fast and flexible denoising, and it does not predict the noise. In this architecture, a reversible downsampling operator and a tunable noise level map transfer the input image to four-images that will enter the CNN layers. In the end, we received four denoised subimages that will be upsampled to form the output image. The experiments have shown that for AWGN removal, the DnCNN is better for low noise levels, and FDDNet is slightly better when the noise level increases.
The robustness is one of the main properties in digital watermarking. Each watermarking scheme survives a specific type of attacks based on the target application. It is the first time that we propose Deep Learning to attack watermarked images, and it could be a harmful type of attack for watermarked images.

3. Watermarking Schemes

There exist a wide range of watermarking schemes for copyright protection and image authentication. They are generally classified as a substitutive class or additive class. Part of it is developed to improve the fidelity, the payload, or the robustness against specific types of attacks. In this paper, we will focus on the STDM watermarking scheme, which belongs to the substitutive class, and the SS watermarking scheme, which belong to the additive class.

STDM [40, 41, 42], a special case of QIM [5], spreads the embedding-induced distortion into a group of samples instead of one sample. The embedding function of STDM is as follows:

$$\begin{aligned} y &= x + (Q_m(x^T p, \Delta) - x^T p)p \\ &= x + \left(round\left(\frac{x^T p - d_m}{\Delta} \right) \Delta + d_m - x^T p \right) p, \end{aligned} \tag{1}$$

where Δ denotes the quantization factor, $round()$ represents the rounding value to the nearest integer, and d_m is the dither level based on the message bit $m \in \{0, 1\}$:

$$d_0 = -\frac{\Delta}{4} \text{ and } d_1 = \frac{\Delta}{4}. \tag{2}$$

The embedded message is extracted using the minimum distance decoder:

$$\hat{m} = arg \min_{m \in \{0,1\}} \mid y^T p - Q_m(y^T p, \Delta) \mid . \tag{3}$$

On the other hand, SS watermarking scheme could be applied in two ways, specifically with additive SS and multiplicative SS [3, 43, 4]. In the additive form, the watermarked image Y is formed by adding the watermark w to the original image X as:

$$Y_i = X_i + \alpha w_i, \tag{4}$$

where α is the scaling factor. In the multiplicative form, the watermark is embedded as:

$$Y_i = X_i(1 + \alpha w_i). \tag{5}$$

4. Convolutional Neural Network

Deep Learning (DL) is a subset of machine learning, inspired by the function and the structure of the brain [44]. DL architectures have been applied in many fields, such as computer vision, image analysis, audio recognition, and image classification [45]. Varieties of DL architectures such as Convolutional Neural Network (CNN) have been studied and used by researchers for a special use case data [15, 16, 17, 18]. CNN model is constructed with input layers, output layers, and hidden layers in between. The major components, as shown in Fig.1 are the convolution layers, pooling or subsampling layers, activation functions, and fully connected layers.

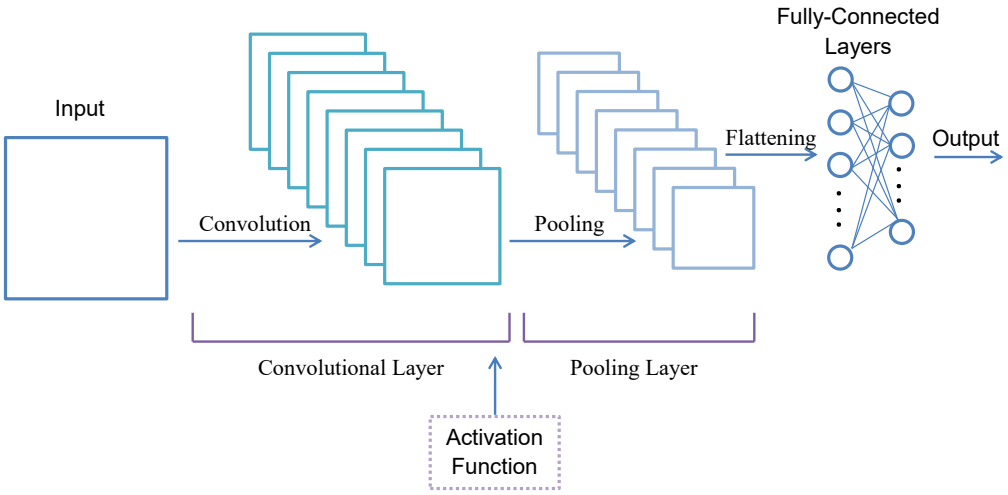

Figure 1: Convolutional Neural Network architecture.

CNN takes an input image, which passes through multiple convolutions and subsampling layers. Each convolution layer includes a series of filters that are presented as matrix numbers. A convolution product will be applied between these filters and previous image matrix to extract important features known as output channels maps. After that, the pooling layers will reduce the dimension of the input map and retain the important information. Max pooling technique is one of the techniques of subsampling, which returns the maximum value from a block. Besides, activation functions such as RELU (Rectified Linear Unit) are usually employed to introduce the non-linearity in the network. The RELU function round the negative values to zero. Noting that, other non-linear functions could be used, such as sigmoid and hyperbolic tangent denoted as Tanh. Also, the Batch normalization could be used when training the network to reduce the overfitting, and decrease the learning time.
CNN models are dominant in many computer vision tasks and have accomplished startling achievements across a variety of domains, such as face recognition, image classification, self-driving cars, and many more.

5. Fully Convolutional Neural Network based Denoising

Nowadays, several types of noise affect the visual quality of digital images. A deep network for image denoising can deal with different kind of noises such

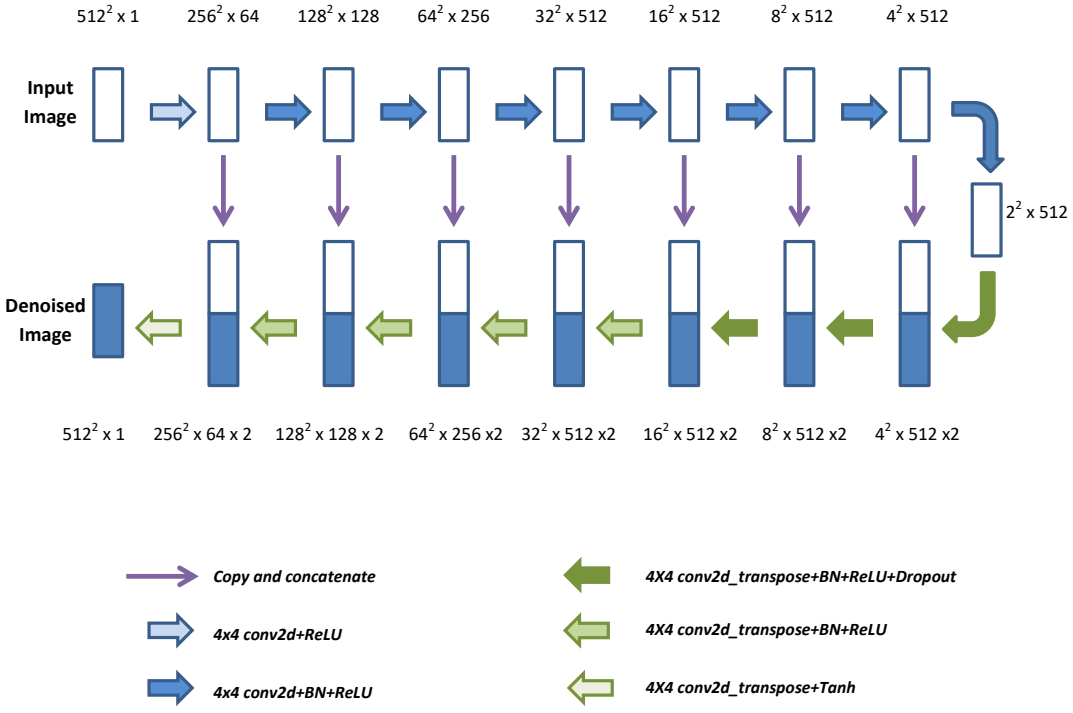

Figure 2: Schematic diagram of the Fully Convolutional Neural Network based Denoising [19].

as speckle noise, and Additive White Gaussian Noise (AWGN)[19]. In this network, the considered generator is the U-Net [46] version that corresponds to an encoder-decoder, which has skip connections between the mirrored layers of the encoder and decoder as shown in Fig. 2. In this figure, each white box denotes a set of feature maps arisen from an encoding layer, and each blue box expresses a set of feature maps arisen from a decoding layer.
Notable features are extracted using the encoder, to remove the noise and preserve the detailed structure of the image simultaneously. Besides, the decoder recovers successive image details and provides a clean version of the noisy image. The skip connections transport directly the information from the encoder layers to its corresponding decoder layers, to share the low-level information between the noisy image and the clean image. Different levels of details are recollected using the skip connections to be used during the reconstruction of the output clean image. The interesting part of symmet-

ric skip connections is that they improve image recovery and facilitate the training of the network. Also, the vanishing gradient problem is solved using the skip connections by directly back-propagating the signal.
The first layer of the encoder is a Convolution-ReLU layer, and the other layers are Convolution-BatchNorm-ReLU layers. The decoder is constituted of transpose Convolution-BatchNorm-ReLU layers with a dropout rate of 50%, and of transpose Convolution-BatchNorm-ReLU layers without dropout. In the end, the denoised image is obtained by a transpose Convolution-Tanh layer.
The operations of pooling and un-pooling are not used in this network, because the aggregation will induce some losses of details, and this would be unsuitable in the image denoising context. Noticed that the convolutions and de-convolutions use 4×4 kernels with stride 2. In this case, each encoder and decoder layer will produce feature maps that are downsampled and upsampled by a factor of 2. The size and number of feature maps are provided in each box in the encoding and decoding part as shown in Fig. 2. For example, let us take the case $4^2 \times 512 \times 2$ in the decoder. This means that we have 1024 maps of size 4^2. 512 maps out of 1024 are higher resolution features map copied from the encoder, and the other 512 are the result of the decoding of the bottleneck layer.
Each operation of the convolution is implemented using the respective TensorFlow module. The network is trained during 50 epochs using Adam optimizer [47] because it provides a faster training convergence of the encoder-decoder network. The initial learning rate for adam is equal to 0.0002, and the training time is about 5 hours. To notes that the computations were completed on an NVIDIA Tesla Titan X GPU.
The loss function has a major impact during the training process. The choice of the loss function is usually based on the $L1$ norm or the $L2$ norm, which is the popular option. The main objective of image denoising is to amend the visual quality of images, and the $L2$ norm is not correlated with this desirable objective, as mentioned by Zhao et al. [48]. $L2$ does not capture the intricate characteristics of the Human Visual System (HVS) that depend on local structure, contrast, and luminance. However, Zhao et al. [48] shown that a combination of $L1$–norm with MS-SSIM achieved the best results. Therefore, we independently came up with the same idea by combining the $L1$ norm and the Structural Similarity (SSIM) index, denoted by $\varsigma^{L1+SSIM}$,

which are defined as:

$$\zeta^{L1}(x,y) = \frac{1}{N}\sum_{p\in P}|x(p)-y(p)|, \tag{6}$$

$$\zeta^{SSIM}(x,y) = 1 - \frac{1}{N}\sum_{p\in P}\frac{2\mu_x\mu_y + c_1}{\mu_x^2+\mu_y^2+c_1}\cdot\frac{2\sigma_{xy}+c_2}{\sigma_x^2+\sigma_y^2+c_2}, \tag{7}$$

where N is the number of pixels p in the patch P. x denotes the noisy image, and y denotes the clean image. σ and μ represent the standard deviation and the means that depend on a pixel p, which are computed using a Gaussian filter with standard deviation σ_G. c_1 and c_2 are two constants $\ll 1$. σ_{xy} can be estimated as [49]:

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i-\mu_x)(y_i-\mu_y). \tag{8}$$

This fully convolutional network was introduced for image denoising. It can remove AWGN and multiplicative speckle noise [19], and it could be a harmful type of attack for watermarked images.

6. Evaluation

STDM and SS watermarking schemes are examined against the Fully Convolutional Neural Network, which can be considered as denoising attack (FCNNDA). A subset of 10000 gray-scale images of 512×512 pixels was used as a data set provided by the BOSS database [50].
For this type of attack, 2000 images are used to train the network during 50 epochs, and 500 remaining images are used during the test. The watermarked images were also tested against different attacks such as salt&pepper, Median filtering, Gaussian filtering, Average filtering, and Wiener filtering. This comparison will show the quality and robustness variation against each type of attack. Structural Similarity Index Measure (SSIM) is used to compare the quality of the watermarked images after applying the attacks. Bit Error Rate (BER) is used to compare the level of robustness when a watermark in form of bits is embedded in the original image. Normalized Correlation (NC) is used to compare the level of robustness for a binary watermark. Normalized Cross-Correlation (NCC) is used to measure the level of robustness of gray-scale watermark.

Also, the Standard Deviation (SD) is used to show the variation from the mean value.

6.1. STDM and DCT based watermarking

The watermarks are embedded in the original images based on STDM and DCT transform. Each image is divided into 16×16 blocks of pixels, and the DCT transform is applied on each block to get the DCT coefficients. The zigzag scanned order is used to select the middle-frequency components, through which the watermark bit is embedded using the STDM watermarking scheme. The embedding rate is 1/256, which allowed the embedding of 1024 bit into each image.
STDM is evaluated against FCNNDA with three different scenarios:

- Scenario 1: Embedding binary image watermark in the original image.
- Scenario 2: Embedding gray-scale watermark in the original image. In this part, each gray-scale value is transformed to the binary form of 8 bits, and each bit is embedded into 1 block of the DCT transform. In total, 1024 bits could be embedded into each image. Therefore, we need 128 gray-scale values to be embedded in each image.
- Scenario 3: Embedding identical redundant bits (0 or 1) in the original image. In this part, the robustness is tested in term of BER, and we have also computed the percentage of faulty extracted bits using a voting algorithm.

For a fair evaluation, the robustness of watermarked schemes is usually tested using watermarked images with uniform fidelity [51, 52, 53]. Therefore, the quantization step size Δ was selected with a value close to 80 to get watermarked images with a fixed SSIM close to 0.986. All the results presented in the graphs and tables are the average results for 500 images extracted from the BOSS database [50].

6.1.1. Scenario 1

In the first scenario, a binary image watermark such as the one presented in Fig.3 of size 32×32 is embedded in the original image of size 512×512 based on STDM and DCT. After that, FCNNDA and other types of attacks are applied to the watermarked images, to compare the quality and robustness levels. Table 1 shows the level of robustness and the quality of

Table 1: Robustness and quality of the attacked images, when binary image watermarks are embedded in the original images with STDM-DCT watermarking (Scenario 1).

Attacks	**NC (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.951 (0.011)	0.868 (0.021)
Salt&Pepper (d=0.01)	0.890 (0.013)	0.770 (0.033)
Gaussian filtering (5×5)	0.998 (0.003)	0.976 (0.005)
Median filtering (3×3)	0.966 (0.027)	0.907 (0.051)
Median filtering (5×5)	0.870 (0.065)	0.813 (0.102)
Wiener filtering (3×3)	0.994 (0.008)	0.929 (0.037)
Wiener filtering (5×5)	0.917 (0.047)	0.861 (0.077)
Average filtering (3×3)	0.941 (0.028)	0.882 (0.005)
Average filtering (5×5)	0.820 (0.064)	0.775 (0.107)
FCNNDA	0.651 (0.142)	0.976 (0.007)

Figure 3: A binary image watermark of size 32×32.

the attacked images. STDM has good robustness against the additive noise and the filtering attacks when the density of the noise is low, or the filters window size is small. But the NC decreases when the noise density or the filters window size increases. In parallel, the quality of the attacked images is also affected after applying such type of attacks.

As shown in Table 1, the watermarks are affected against the Average filtering (5×5), with a NC average close to 0.820. But the quality of the attacked images is also affected with an SSIM average close to 0.775. On the other hand, the FCNNDA disturbs almost all the watermarks, with a NC average close to 0.651, while preserving a good quality of the attacked images (SSIM = 0.976). The Standard Deviation (SD) values of NC and SSIM are low, which means that most of the values are close to the average. Fig.4 presents

(a)Watermarked Image. (b)Salt & Pepper (d=0.01). (c)Median Filtering (5×5). (d)Wiener Filtering (5×5). (e)Average Filtering (5×5). (f)FCNNDA.

Figure 4: Quality distortion of attacked image, when binary watermark is embedded in the original image with STDM-DCT watermarking (Scenario 1).

(a)Salt& Pepper (d=0.01). (b)Gaussian Filtering (5×5). (c)Median Filtering (5×5). (d)Wiener Filtering (5×5). (e)Average Filtering (5×5). (f)FCNNDA.

Figure 5: Extracted binary image watermark after applying the different types of attacks.

the quality effect of those attacks on the watermarked image. We have selected the Lena photo out of the 500 images in this part because it is widely used in image processing due to its detail, flat regions, shading, and texture. Fig.5 shows the extracted binary image watermark after applying the different types of attacks. With all the attacks except for FCNNDA, the binary image watermark could be reconstructed using an algorithm of denoising. With FCNNDA, the extracted binary image watermark is relatively different from the extracted one. Therefore, it would be difficult and somehow impossible to reconstruct the extracted binary image watermark after the FCNNDA attack.

The Histograms in Fig.6 show the absolute difference between the pixels values of the original images, watermarked images, and attacked images by Average filtering (5×5), and by FCNNDA. 96% of the absolute difference between the pixels values of the watermarked images and the original images

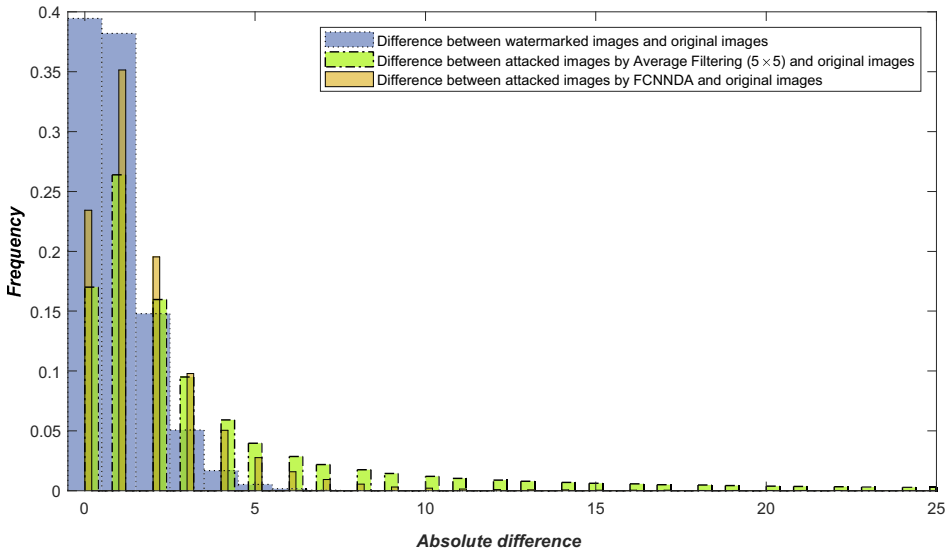

Figure 6: Comparing the pixels values of the original images to the watermarked images, and the attacked images by Average filtering (5×5) and FCNNDA, when the watermark is a binary image of size 32×32, and the watermarking scheme is STDM-DCT.

are distributed between 0 and 3. 89% of the absolute difference between the pixels values of the attacked images by FCNNDA and the original images are distributed between 0 and 3. 68% of the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images are distributed between 0 and 3. The attacked images by FCNNDA are thus closer to the original images comparing to the attacked images by Average filtering (5×5). Noting that, we have only compared the FCNNDA to the Average filtering attack because they have the lowest NC values.
Moreover, we have visually interpreted the absolute difference between the pixels values of the original images, the watermarked images, and attacked images by FCNNDA and Average filtering. The results are presented in Fig.7, where the watermark was embedded into the original image "Lena" using different projection vectors. We have used different projection vectors with STDM during the test to study their impact on the effectiveness of the FCNNDA attack. The first and the second columns in Fig.7 contains two watermarked images "Lena" with different projection vectors. The third and the fourth columns present zooming to the same block of the first two images for more details. The first line presents the absolute difference between the pixels values of the original image and the watermarked image. We can

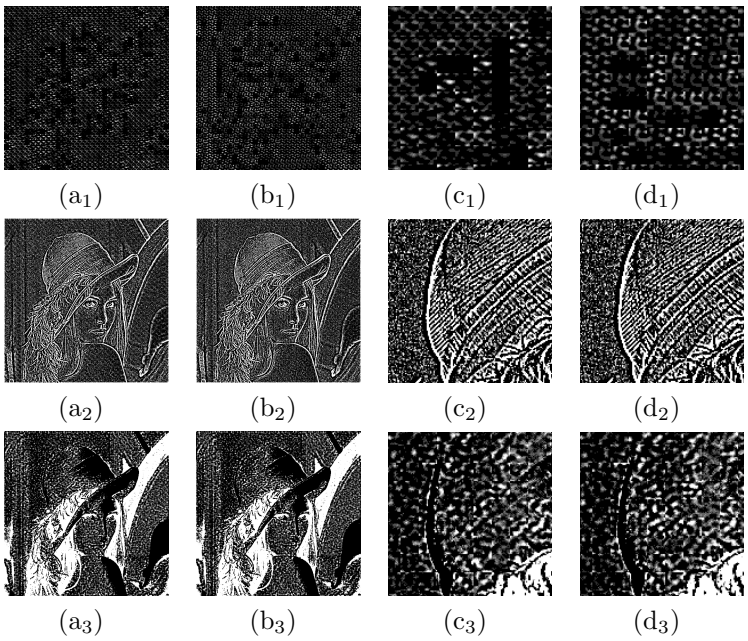

Figure 7: Absolute difference (a_1) between the pixels values of the original images and the watermarked images, (a_2) between the pixels values of the original images and attacked images by Average filtering (5×5), (a_3) between the pixels values of the original images and attacked images by FCNNDA, (b_i) presents the absolute difference with different projection vector for STDM, (c_i) and (d_i) present the details for (a_i) and (b_i) respectively.

recognize the patches in the images, and we can notice that the watermark bits are embedded everywhere in the image. The second line shows the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images. All the edges are changed due to the average filtering. The third line shows the absolute difference between the pixels values of the attacked images by FCNNDA and the original images. We can distinguish how the patches are changed in the images. The darken parts, and the brighten parts of the images are also changed compared to the original one. This variation destroys the watermarks that have been embedded in the original images.

As shown in this section, the FCNNDA attack outperforms the other types of attacks. During the training process of the FCNNDA network, notable features are extracted from the encoder and shared with de decoder using the skip connections to remove the noise and preserve the detailed structure

Table 2: Robustness and quality of the attacked images, when gray-scale watermark is embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 2).

Attacks	**NCC (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.796 (0.073)	0.868 (0.021)
Salt&Pepper (d=0.01)	0.612 (0.086)	0.769 (0.032)
Gaussian filtering (5×5)	0.998 (0.005)	0.976 (0.005)
Median filtering (3×3)	0.878 (0.128)	0.907 (0.052)
Median filtering (5×5)	0.626 (0.234)	0.813 (0.102)
Wiener filtering (3×3)	0.975 (0.043)	0.928 (0.037)
Wiener filtering (5×5)	0.609 (0.212)	0.861 (0.077)
Average filtering (3×3)	0.836 (0.136)	0.881 (0.055)
Average filtering (5×5)	0.418 (0.225)	0.774 (0.107)
FCNNDA	0.259 (0.198)	0.974 (0.008)

of the image. The major impact during the training process is based on the Loss function that has reflected the visual quality and fulfills the desired objective.

6.1.2. Scenario 2

In the second scenario, a gray-scale watermark is embedded in each original image. Each gray-scale value is converted to the binary form of 8 bits, and each bit is embedded into 1 block of the DCT transform. The robustness and the quality of the attacked images are tested in term of NCC and SSIM. 1024 bits could be embedded into each image. Hence, 128 gray-scale values are embedded to each image based on STDM and DCT.

As shown in Table 2, the watermarks are affected against the Average filtering (5×5), with a NCC average close to 0.418. But the quality of the attacked images is also affected with an SSIM average close to 0.774. On the other hand, the FCNNDA destroys the gray-scale watermark with a NCC average close to 0.259, while preserving a good quality of the attacked images (SSIM = 0.974). The gray-scale watermark is highly affected comparing to the binary image watermark because the gray-scale value is converted to

(a)Watermarked Image. (b)Average Filtering (5×5). (c)FCNNDA.

Figure 8: Quality distortion of attacked images, when gray-scale watermark is embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 2).

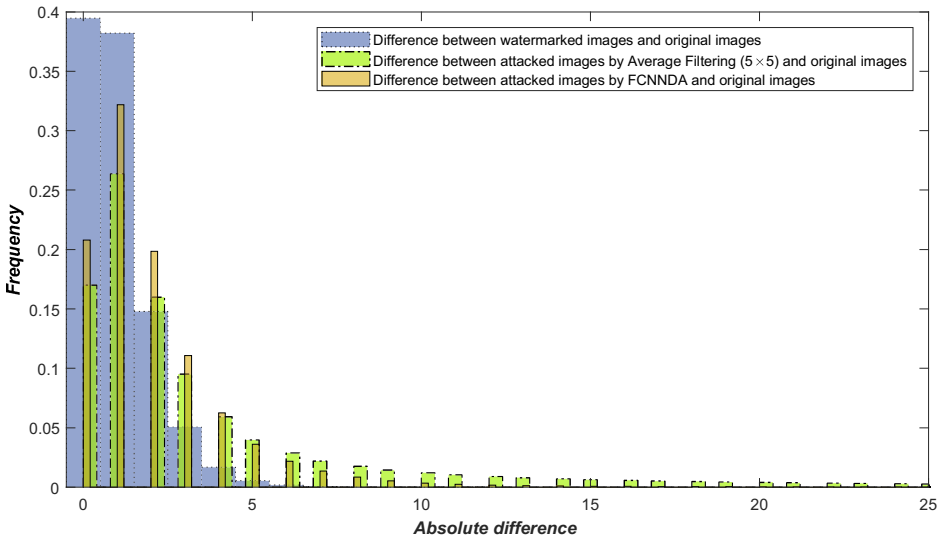

Figure 9: Comparing the pixels values of the original images to the watermarked images, and the attacked images by FCNNDA and Average filtering (5×5), when gray-scale watermark is embedded per image, and STDM with DCT is used as watermarking scheme.

the binary form of 8 bits. In this case, if one bit out of 8 bits is changed after the attack, the gray-scale value will also be changed. The quality effect

of Average filtering and FCNNDA attacks on the watermarked image are presented in Fig.8. The absolute difference between the pixels values of the original images, the watermarked images, and attacked images (by FCNNDA and Average filtering) are presented in the histograms in Fig.9. 95% of the absolute difference between the pixels values of the watermarked images and the original images are distributed between 0 and 3. 87% of the absolute difference between the pixels values of the attacked images by FCNNDA and the original images are distributed between 0 and 3. 68% of the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images are distributed between 0 and 3. The attacked images by FCNNDA are closer to the original images comparing to the attacked images by Average filtering (5×5).

Table 3: Percentage of faulty bits and quality of the attacked images, when an identical bit is embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 3).

Attacks	**% Faulty Bits**	**BER (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0	0.064 (0.007)	0.876 (0.003)
Salt&Pepper (d=0.01)	0	0.132 (0.011)	0.779 (0.004)
Gaussian filtering (5×5)	0	0.007 (0.002)	0.975 (0.008)
Median filtering (3×3)	0	0.049 (0.002)	0.896 (0.004)
Median filtering (5×5)	0	0.212 (0.011)	0.781 (0.001)
Wiener filtering (3×3)	0	0.017 (0.001)	0.913 (0.005)
Wiener filtering (5×5)	0	0.160 (0.003)	0.827 (0.003)
Average filtering (3×3)	0	0.120 (0.002)	0.864 (0.003)
Average filtering (5×5)	0	0.261 (0.011)	0.731 (0.008)
FCNNDA	48	0.481 (0.149)	0.970 (0.011)

(a)Watermarked Image. (b)Average Filtering (5×5). (c)FCNNDA.

Figure 10: Quality distortion of attacked images, when an identical bit is embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 3).

6.1.3. Scenario 3

In the third scenario, a sequence of 1024 length 0 or 1 identical bit is embedded in each original image. The robustness and the quality of the attacked images are tested in term of BER and SSIM. We have also computed the percentage of faulty extracted bits using a voting algorithm. If the majority of extracted bits per image are wrong, the voting value will increase by 1, and in this way, the percentage value will increase as well. By this way, we could determine if the identical redundant bit could be extracted without error after applying the attacks. As shown in Table 3, the majority of extracted bits per image are correct. For that the total result of voting algorithm was equal to 0% faulty bits for all the attacks excepting FCNNDA. Conversely, we have got 48% faulty bits when applied the FCNNDA attack; with 48% of the images, the majority of extracted bits per image are incorrect. The average of BER is close to 0.481, and the quality of the attacked images has an SSIM value close to 0.970. The quality effect of Average filtering and FCNNDA attacks on the watermarked image are presented in Fig.10. The absolute difference between the pixels values of the original images, the watermarked images, and attacked images (by FCNNDA and Average filtering) are presented in the histograms in Fig.11. 96% of the absolute difference between the pixels values of the watermarked images and the original images

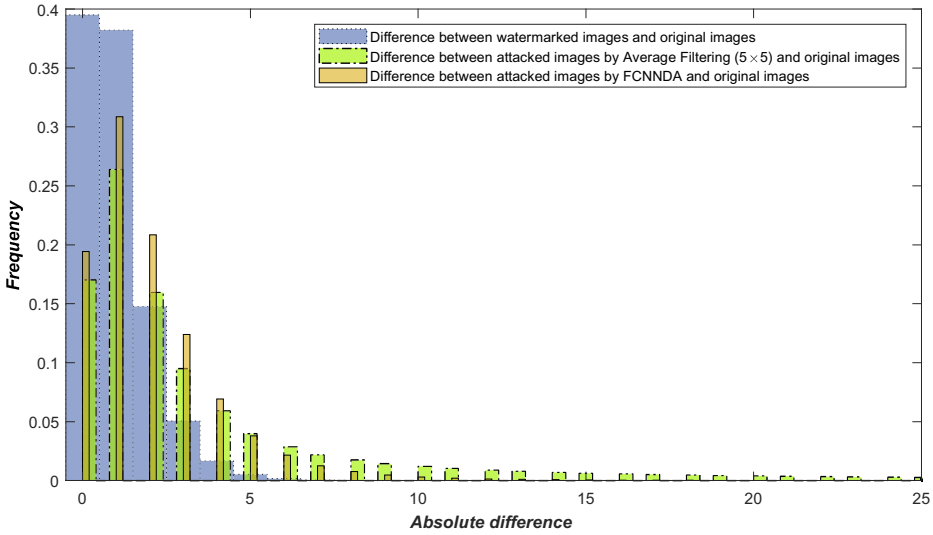

Figure 11: Comparing the pixels values of the original images to the watermarked images, and the attacked images by FCNNDA and Average filtering (5×5), when an identical bit is embedded per image, and STDM with DCT is used as watermarking scheme.

are distributed between 0 and 3. 85% of the absolute difference between the pixels values of the attacked images by FCNNDA and the original images are distributed between 0 and 3. 69% of the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images are distributed between 0 and 3. The attacked images by FCNNDA are closer to the original images comparing to the attacked images by Average filtering (5×5).
The extracted identical bit could be different from the embedded one with a probability close to 0.5 after applying the FCNNDA attack. This is a high value comparing to the other type of attacks, where the value of the faulty extracted bits was equal to 0.

6.2. SS and DWT-SVD based watermarking

In this part of the experiments, the SS watermarking scheme was tested against the FCNNDA attack. SVD and DWT were widely used with SS to increase the level of robustness and fidelity [54, 55, 56]. Therefore, we have embedded the watermark in the original images using the DWT-SVD as follows:

- Perform 3-level DWT onto the cover image to get the four sub-bands

(LL3, LH3, HL3, and HH3).

- Apply the SVD to LL3 sub-band ($A = USV^T$), and the watermark ($A_w = U_w S_w V_w^T$).
- Modify the singular values of LL3 by adding the singular values of the watermark as: S'= S+ αS_w. Alpha is the scaling factor.
- Perform the modified LL3: $LL^{new} = US'V^T$.
- Finally, apply the inverse DWT to obtain the watermarked image A_w.

The watermark was extracted as follows:

- Perform 3-level DWT onto the watermarked image A_W^* to decompose it into four sub-bands (LL3, LH3, HL3, and HH3).
- Apply SVD to LL3: $A^* = U^* S^* V^{*T}$.
- Compute: S_W^*= $\frac{S^*-S}{\alpha}$.
- Get the watermark as: $S_W^* = U_w S_W^* V_w^T$.

The tested images are watermarked using a uniform fidelity. The scaling factor α was selected with a value close to 0.15 to get watermarked images with fixed SSIM values close to 0.986.
SS is evaluated against FCNNDA with three different scenarios:

- Scenario 1: Embedding gray-scale watermark of size 64×64 in the original image.
- Scenario 2: Embedding binary image watermark of size 64×64 in the original image. (To apply the SVD on the binary watermark, we have multiplied the binary values by 255).
- Scenario 3: Embedding identical redundant bits (0 or 1) in the original image. The redundant bits are grouped in a matrix of 64×64, and the bits 0 are replaced by 64, and the bits 1 are replaced by 192, to apply the SVD on the matrix.

Table 4: Robustness and quality of the attacked images, when gray-scale watermarks are embedded in the original images, and SS with DWT-SVD is used as watermarking scheme (Scenario-1).

Attacks	**NCC (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.952 (0.064)	0.879 (0.018)
Salt&Pepper (d=0.01)	0.875 (0.143)	0.778 (0.032)
Gaussian filtering (5×5)	0.985 (0.021)	0.984 (0.006)
Median filtering (3×3)	0.968 (0.044)	0.914 (0.052)
Median filtering (5×5)	0.861 (0.118)	0.816 (0.102)
Wiener filtering (3×3)	0.969 (0.037)	0.934 (0.038)
Wiener filtering (5×5)	0.851 (0.127)	0.862 (0.078)
Average filtering (3×3)	0.888 (0.128)	0.889 (0.056)
Average filtering (5×5)	0.518 (0.227)	0.776 (0.107)
FCNNDA	0.136 (0.152)	0.986 (0.034)

Figure 12: Gray-scale watermark of size 64×64.

Structural Similarity Index Measure (SSIM) is used to compare the quality of the watermarked images after applying the attacks. Bit Error Rate (BER) is used to compare the level of robustness when a watermark in form of bits is embedded in the original image. Normalized Correlation (NC) is used to compare the level of robustness for a binary watermark. Normalized Cross-Correlation (NCC) is used to compare the level of robustness of gray-scale watermark. All the results presented in the tables are the average results for 500 images extracted from the BOSS database [50].

6.2.1. Scenario 1

In the first scenario, a gray-scale watermark like the cameraman presented in Fig.12 of size 64×64 is embedded in the original image using the SS and

(a)Watermarked Image. (b)Average Filtering (5×5). (c)FCNNDA.

Figure 13: Quality distortion of attacked images, when gray-scale watermark is embedded in the original image, and SS with DWT-SVD is used as watermarking scheme (Scenario-1).

(a)Salt& Pepper (d=0.01). (b)Gaussian Filtering (5×5). (c)Median Filtering (5×5). (d)Wiener Filtering (5×5). (e)Average Filtering (5×5). (f)FCNNDA.

Figure 14: Extracted gray-scale watermark after applying the different types of attacks.

DWT-SVD based watermarking. After that, FCNNDA and the other types of attacks are applied to the watermarked images, to compare the quality and robustness levels.

As shown in Table 4, the extracted watermark is affected when the Average filtering (5×5) is applied to the watermarked images, where the average of NCC value is close to 0.518. In parallel, the quality of the attacked images is also affected, where the average of SSIM values is close to 0.776. Conversely, the extracted watermark is highly affected by the FCNNDA attack, where the average of NCC is close to 0.152. Also, the quality of the attacked images is preserved, where the average of SSIM values is close to 0.984. The quality

Table 5: Robustness and quality of the attacked images, when binary image watermarks are embedded in the original images, and SS with DWT-SVD is used as watermarking scheme (Scenario-2).

Attacks	**NC (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.989 (0.051)	0.879 (0.017)
Salt&Pepper (d=0.01)	0.957 (0.125)	0.778 (0.32)
Gaussian filtering (5×5)	0.999 (0.001)	0.986 (0.006)
Median filtering (3×3)	0.992 (0.047)	0.914 (0.051)
Median filtering (5×5)	0.898 (0.114)	0.816 (0.102)
Wiener filtering (3×3)	0.996 (0.011)	0.934 (0.038)
Wiener filtering (5×5)	0.896 (0.097)	0.862 (0.077)
Average filtering (3×3)	0.941 (0.081)	0.888 (0.056)
Average filtering (5×5)	0.589 (0.207)	0.776 (0.107)
FCNNDA	0.211 (0.157)	0.985 (0.007)

effect of Average filtering and FCNNDA attacks on the watermarked image are presented in Fig.13. FCNNDA destroys the embedded watermark while preserving a good quality of the watermarked image.
Fig.14 shows the extracted gray-scale watermark after applying the different types of attacks. With all the attacks except for FCNNDA, the gray-scale watermark could be repaired using an algorithm of denoising or filtering. With FCNNDA, the extracted gray-scale watermark is relatively different from the extracted one. Hence, it would be tricky and somehow impossible to reconstruct the extracted gray-scale watermark after the FCNNDA attack.

6.2.2. Scenario 2

In the second scenario, a binary image watermark of size 32×32 is embedded in the original image of size 512×512 based on SS and DWT-SVD. SVD is applied to the binary watermark after multiplying the binary values by 255.
Table 5 shows the level of robustness and the quality of the attacked images, when binary watermarks are embedded in the original images. In this sce-

(a)Watermarked Image. (b)Watermarked Image (Zoom) (c)FCNNDA (a)FCNNDA (Zoom)

Figure 15: Quality distortion of FCNNDA, when Binary watermark image is embedded in the original image, and SS with DWT-SVD is used as watermarking scheme (Scenario-2).

(a)Salt& Pepper (d=0.01). (b)Gaussian Filtering (5×5). (c)Median Filtering (5×5). (d)Wiener Filtering (5×5). (e)Average Filtering (5×5). (f)FCNNDA.

Figure 16: Extracted binary image watermark after applying the different types of attacks.

nario SS watermarking scheme achieved good robustness against the additive noise and the filtering attacks. But the NC decreases with the Average filtering (5×5) attack with a NC average close to 0.589. But the quality of the attacked images is also affected with an SSIM average close to 0.776. On the other hand, the FCNNDA disturbs almost all the binary image watermarks, with a NC average close to 0.211, while preserving the quality of the attacked images (SSIM = 0.985). The SD values of NC and SSIM are low, which means that most of the values are close to the average. Fig.15 presents the quality effect of FCNNDA attack on the watermarked image.
Fig.16 shows the extracted binary image watermark after applying the different types of attacks. With all the attacks except for FCNNDA, the binary image watermark could be reconstructed using an algorithm of denoising. With FCNNDA, the extracted binary image watermark is relatively different from the extracted one. Therefore, it would be difficult and somehow impossible to reconstruct the extracted binary image watermark after the FCNNDA attack.

Table 6: Percentage of faulty bits and quality of the attacked images, when an identical bit is embedded per image, and SS with DWT-SVD is used as watermarking scheme (Scenario-3).

Attacks	**% Faulty Bits**	**BER (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0	0.045 (0.182)	0.876 (0.019)
Salt&Pepper (d=0.01)	0	0.051 (0.192)	0.775 (0.031)
Gaussian filtering (5×5)	0	0.033 (0.151)	0.984 (0.007)
Median filtering (3×3)	0	0.013 (0.099)	0.912 (0.051)
Median filtering (5×5)	0	0.034 (0.162)	0.815 (0.103)
Wiener filtering (3×3)	0	0.042 (0.161)	0.932 (0.037)
Wiener filtering (5×5))	0	0.051 (0.183)	0.860 (0.076)
Average filtering (3×3)	0	0.032 (0.098)	0.886 (0.055)
Average filtering (5×5)	0	0.072 (0.234)	0.774 (0.106)
FCNNDA	**32**	0.342 (0.241)	0.987 (0.008)

6.2.3. Scenario 3

In the third scenario, a sequence of 1024 length 0 or 1 identical bit is embedded in the original image. The redundant bits are grouped in a matrix of 64×64, and the bits 0 are replaced by 64, and the bits 1 are replaced by 192, to apply the SVD on the matrix. The robustness and the quality of the attacked images are tested in term of BER and SSIM. We have also computed the percentage of faulty extracted bits using a voting algorithm. If the majority of extracted bits per image are wrong, the voting value will increase by 1, and in this way, the percentage value will increase as well. Based on this scenario, we could determine if the identical redundant bit could be extracted without error after applying the attacks. As shown in Table 6, the majority of extracted bits per image are correct. For that, the total result of the voting algorithm was equal to 0% faulty bits for all the

(a)Watermarked Image. (b)Watermarked Image (Zoom) (c)FCNNDA (a)FCNNDA (Zoom)

Figure 17: Quality distortion of FCNNDA, when identical bit is embedded per image, and SS with DWT-SVD is used as watermarking scheme (Scenario-3).

attacks excepting FCNNDA. Conversely, we have got 32% faulty bits when the FCNNDA attack is applied; with 32% of the images, the majority of extracted bits per image was incorrect. The average of BER is close to 0.342, and the quality of the attacked images has an SSIM value close to 0.987. The identical embedded bit has a probability close to 0.3 to be extracted with error after applying the FCNNDA attack. This is a high value comparing to the other type of attacks, where the faulty extracted bits was equal to 0%. The quality effect of watermarked image and FCNNDA attack on the watermarked image are presented in Fig. 17.

7. Conclusion and Future Work

In this paper, digital image watermarking are evaluated against a Fully Convolutional Neural Network Denoising Attack (FCNNDA). STDM and SS watermarking schemes are examined against FCNNDA using different scenarios in the frequency domain. Several types of watermarks are embedded during the test such as binary watermarks, one redundant bit, and gray-scale watermarks. FCNNDA was also compared to other types of attacks to show the difference in term of quality and robustness. The experimental results confirmed that the FCNNDA could be considered as a harmful attack. FCNNDA outperforms the other types of attacks because it destroys the watermarks while preserving a good quality of the attacked images.
As for future work, we plan to integrate deep learning with digital watermarking to enhance the robustness against such type of attacks.

Acknowledgements

This work is partially funded with support from the Labex ACTION program (contract ANR-11-LABX-01-01), the National Council for Scientific Research in Lebanon CNRS-L, the Hubert Curien CEDRE programme, and the Agence Universitaire de la Francophonie AUF-PCSI programme. We also thanks the mesocentre of Franche-Comté for the computing facilities.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] I. Cox, M. Miller, J. Bloom, J. Fridrich, T. Kalker, Digital watermarking and steganography, Morgan Kaufmann, 2007.

[2] I. J. Cox, J. Kilian, T. Leighton, T. Shamoon, Secure spread spectrum watermarking for images, audio and video, in: Proceedings of 3rd IEEE International Conference on Image Processing, volume 3, 1996, pp. 243–246 vol.3. doi:10.1109/ICIP.1996.560429.

[3] I. J. Cox, J. Kilian, F. T. Leighton, T. Shamoon, Secure spread spectrum watermarking for multimedia, IEEE Transactions on Image Processing 6 (1997) 1673–1687. doi:10.1109/83.650120.

[4] H. S. Malvar, D. A. F. Florencio, Improved spread spectrum: a new modulation technique for robust watermarking, IEEE Transactions on Signal Processing 51 (2003) 898–905. doi:10.1109/TSP.2003.809385.

[5] B. Chen, G. W. Wornell, Quantization index modulation: a class of provably good methods for digital watermarking and information embedding, IEEE Transactions on Information Theory 47 (2001) 1423–1443. doi:10.1109/18.923725.

[6] B. Chen, G. W. Wornell, Quantization index modulation methods for digital watermarking and information embedding of multimedia, Journal of VLSI signal processing systems for signal, image and video technology 27 (2001) 7–33.

[7] V. M. Potdar, S. Han, E. Chang, A survey of digital image watermarking techniques, in: INDIN '05. 2005 3rd IEEE International Conference on Industrial Informatics, 2005., 2005, pp. 709–716. doi:10.1109/INDIN.2005.1560462.

[8] G. C. Langelaar, I. Setyawan, R. L. Lagendijk, Watermarking digital image and video data. A state-of-the-art overview, IEEE Signal Processing Magazine 17 (2000) 20–46. doi:10.1109/79.879337.

[9] Q. Li, C. Yuan, Y. Zhong, Adaptive DWT-SVD Domain Image Watermarking Using Human Visual Model, in: The 9th International Conference on Advanced Communication Technology, volume 3, 2007, pp. 1947–1951. doi:10.1109/ICACT.2007.358752.

[10] A. K. Singh, M. Dave, A. Mohan, Hybrid Technique for Robust and Imperceptible Image Watermarking in DWT–DCT–SVD Domain, National Academy Science Letters 37 (2014) 351–358. doi:10.1007/s40009-014-0241-8.

[11] A. Graves, A. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, 2013, pp. 6645–6649. doi:10.1109/ICASSP.2013.6638947.

[12] L. A. Gatys, A. S. Ecker, M. Bethge, Image style transfer using convolutional neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2414–2423.

[13] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Fei-Fei, Large-scale video classification with convolutional neural networks, in: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2014, pp. 1725–1732.

[14] Y. Goldberg, Neural network methods for natural language processing, Synthesis Lectures on Human Language Technologies 10 (2017) 1–309.

[15] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: F. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 25, Curran Associates, Inc., 2012, pp. 1097–1105.

[16] M. Rastegari, V. Ordonez, J. Redmon, A. Farhadi, Xnor-net: Imagenet Classification Using Binary Convolutional Neural Networks, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham, 2016, pp. 525–542.

[17] F. Milletari, N. Navab, S. Ahmadi, V-net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation, in: 2016 Fourth International Conference on 3D Vision (3DV), 2016, pp. 565–571. doi:10.1109/3DV.2016.79.

[18] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.

[19] R. Couturier, G. Perrot, M. Salomon, Image Denoising Using a Deep Encoder-Decoder Network with Skip Connections, in: L. Cheng, A. C. S. Leung, S. Ozawa (Eds.), Neural Information Processing, Springer International Publishing, Cham, 2018, pp. 554–565.

[20] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising, IEEE Transactions on Image Processing 26 (2017) 3142–3155. doi:10.1109/TIP.2017.2662206.

[21] K. Zhang, W. Zuo, L. Zhang, Ffdnet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising, IEEE Transactions on Image Processing 27 (2018) 4608–4622. doi:10.1109/TIP.2018.2839891.

[22] K. Zhang, W. Zuo, S. Gu, L. Zhang, Learning Deep CNN Denoiser Prior for Image Restoration, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[23] C. Song, S. Sudirman, M. Merabti, D. Llewellyn-Jones, Analysis of Digital Image Watermark Attacks, in: 2010 7th IEEE Consumer Communications and Networking Conference, 2010, pp. 1–5. doi:10.1109/CCNC.2010.5421631.

[24] H. Tao, L. Chongmin, J. M. Zain, A. N. Abdalla, Robust image watermarking theories and techniques: A review, Journal of applied research and technology 12 (2014) 122–138.

[25] F. Y. Shih, Digital watermarking and steganography: fundamentals and techniques, CRC press, 2017.

[26] V. Licks, R. Jordan, Geometric attacks on image watermarking systems, IEEE multimedia (2005) 68–78.

[27] Enping Li, Huaqing Liang, Xinxin Niu, Blind Image Watermarking Scheme Based on Wavelet Tree Quantization Robust to Geometric Attacks, in: 2006 6th World Congress on Intelligent Control and Automation, volume 2, 2006, pp. 10256–10260. doi:10.1109/WCICA.2006.1714009.

[28] J.-C. Liu, C.-H. Lin, L.-C. Kuo, J.-C. Chang, Robust Multi-scale Full-Band Image Watermarking for Copyright Protection, in: H. G. Okuno, M. Ali (Eds.), New Trends in Applied Artificial Intelligence, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 176–184.

[29] J. Li, Robust image watermarking scheme against geometric attacks using a computer-generated hologram, Applied optics 49 (2010) 6302–6312.

[30] X. He, T. Zhu, G. Yang, A geometrical attack resistant image watermarking algorithm based on histogram modification, Multidimensional Systems and Signal Processing 26 (2015) 291–306. doi:10.1007/s11045-013-0257-0.

[31] S. Fazli, M. Moeini, A robust image watermarking method based on DWT, DCT, and SVD using a new technique for correction of main geometric attacks, Optik 127 (2016) 964 – 972. doi:https://doi.org/10.1016/j.ijleo.2015.09.205.

[32] Q. Li, G. Doerr, I. J. Cox, Spread Transform Dither Modulation using a Perceptual Model, in: 2006 IEEE Workshop on Multimedia Signal Processing, 2006, pp. 98–102. doi:10.1109/MMSP.2006.285276.

[33] S. D. Lin, S.-C. Shie, J. Guo, Improving the robustness of DCT-based image watermarking against JPEG compression, Computer Standards & Interfaces 32 (2010) 54 – 60. doi:https://doi.org/10.1016/j.csi.2009.06.004.

[34] Q. Li, I. J. Cox, Improved Spread Transform Dither Modulation using a Perceptual Model: Robustness to Amplitude Scaling and JPEG Compression, in: 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, volume 2, 2007, pp. II–185–II–188. doi:10.1109/ICASSP.2007.366203.

[35] M. W. Hatoum, R. Darazi, J. Couchot, Blind Image Watermarking using Normalized STDM robust against Fixed Gain Attack, in: 2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET), 2018, pp. 1–6. doi:10.1109/IMCET.2018.8603038.

[36] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising, IEEE Transactions on Image Processing 26 (2017) 3142–3155.

[37] S. Gu, L. Zhang, W. Zuo, X. Feng, Weighted nuclear norm minimization with application to image denoising, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[38] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, IEEE Transactions on Image Processing 16 (2007) 2080–2095.

[39] K. Zhang, W. Zuo, L. Zhang, Ffdnet: Toward a fast and flexible solution for cnn-based image denoising, IEEE Transactions on Image Processing 27 (2018) 4608–4622.

[40] M. W. Hatoum, R. Darazi, J.-F. Couchot, Normalized blind STDM watermarking scheme for images and PDF documents robust against fixed gain attack, Multimedia Tools and Applications 79 (2020) 1887–1919. doi:https://doi.org/10.1007/s11042-019-08242-4.

[41] A. W. Bitar, R. Darazi, J.-F. Couchot, R. Couturier, Blind digital watermarking in PDF documents using Spread Transform Dither Modulation, Multimedia Tools and Applications 76 (2017) 143–161. doi:10.1007/s11042-015-3034-2.

[42] M. W. Hatoum., R. Darazi., J. Couchot., Blind PDF Document Watermarking Robust Against PCA and ICA attacks, in: Proceedings of the

15th International Joint Conference on e-Business and Telecommunications - Volume 1: SECRYPT,, INSTICC, SciTePress, 2018, pp. 420–427. doi:10.5220/0006899605860593.

[43] A. Valizadeh, Z. J. Wang, An Improved Multiplicative Spread Spectrum Embedding Scheme for Data Hiding, IEEE Transactions on Information Forensics and Security 7 (2012) 1127–1143. doi:10.1109/TIFS.2012.2199312.

[44] M. A. Nielsen, Neural networks and deep learning, volume 25, Determination press San Francisco, CA, USA:, 2015.

[45] L. Deng, A tutorial survey of architectures, algorithms, and applications for deep learning, APSIPA Transactions on Signal and Information Processing 3 (2014).

[46] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, in: N. Navab, J. Hornegger, W. M. Wells, A. F. Frangi (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Springer International Publishing, Cham, 2015, pp. 234–241.

[47] D. Kingma, J. Ba, Adam: A Method for Stochastic Optimization In: Proceedings of the 3rd international conference for learning representations (iclr'15), San Diego (2015).

[48] H. Zhao, O. Gallo, I. Frosio, J. Kautz, Loss Functions for Image Restoration With Neural Networks, IEEE Transactions on Computational Imaging 3 (2017) 47–57. doi:10.1109/TCI.2016.2644865.

[49] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, et al., Image quality assessment: from error visibility to structural similarity, IEEE transactions on image processing 13 (2004) 600–612.

[50] P. Bas, T. Filler, T. Pevný, "break Our Steganographic System": The Ins and Outs of Organizing BOSS, in: T. Filler, T. Pevný, S. Craver, A. Ker (Eds.), Information Hiding, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 59–70.

[51] Dong Yu, Lihong Ma, Guoxi Wang, Hanqing Lu, Adaptive spread-transform dither modulation using an improved luminance-masked

threshold, in: 2008 15th IEEE International Conference on Image Processing, 2008, pp. 449–452. doi:10.1109/ICIP.2008.4711788.

[52] W. Wan, J. Liu, J. Sun, X. Yang, X. Nie, F. Wang, Logarithmic Spread-Transform Dither Modulation watermarking Based on Perceptual Model, in: 2013 IEEE International Conference on Image Processing, 2013, pp. 4522–4526. doi:10.1109/ICIP.2013.6738931.

[53] W. Wan, J. Liu, J. Sun, D. Gao, Improved logarithmic spread transform dither modulation using a robust perceptual model, Multimedia Tools and Applications 75 (2016) 13481–13502. doi:10.1007/s11042-015-2853-5.

[54] A. Sverdlov, S. Dexter, A. M. Eskicioglu, Robust DCT-SVD domain image watermarking for copyright protection: Embedding data in all frequencies, in: 2005 13th European Signal Processing Conference, 2005, pp. 1–4.

[55] C. Lai, C. Tsai, Digital Image Watermarking Using Discrete Wavelet Transform and Singular Value Decomposition, IEEE Transactions on Instrumentation and Measurement 59 (2010) 3060–3063. doi:10.1109/TIM.2010.2066770.

[56] N. M. Makbol, B. E. Khoo, Robust blind image watermarking scheme based on Redundant Discrete Wavelet Transform and Singular Value Decomposition, AEU - International Journal of Electronics and Communications 67 (2013) 102 – 112. doi:https://doi.org/10.1016/j.aeue.2012.06.008.