Implementation of Digital Image Watermarking using Raspberry Pi4

Noor Mohammed Abdulwahab **Dr. Nasseer M. Basheer**

Technical Engineering College / Northern Technical University /

Mosul / Iraq
Email: noor.m.alazzawi84@gmail.com

ABSTRACT

Multimedia applications have exploded because of the Internet revolution. As a result, it has become a standard practice to copy, transfer, and share digital data, which has led in the unlawful use of data resources and the hacking of it by unauthorized users.

One of the recommended methods for copyright protection is digital watermarking. This study emphasizes the need of digital image watermarking, which secures images by hiding a watermark image in it, to show its legal ownership.

This study proposed a technique for embedding and retrieving a watermark image. For further protection, the watermark image is encrypted with a secret key and implemented on the Raspberry Pi4 platform.It gave a good robustness against many attacks. Python has been used to perform and implement this work.

Keywords: Copyright protection, (LSB) least significant bit embedding, watermarking (PSNR) peak signal to noise ratio, Raspberry Pi4.

1. Introduction

The technique of watermarking is used for hiding data or information inside digital multimedia. This study is focused particularly on the digital image watermarking. Digital watermarking gets common, especially when adding watermarks that cannot be detected, such as author's information or copyright information [1]. For this reason, robust methods were developed so that they can protect the rights proprietary of the owners of data. In the embedding process of digital image watermarking, a digital image - which is named a watermark - is embedded inside a host image, the resulted image which is called watermarked image is saved or transmitted. After that extracting the watermark image enables the owners of data to confirm the authenticity and ownership of their object [2].

1.1 basics of watermarking technique

Insertion of a watermark should be done in such a way that the watermark can be completely recovered from the watermarked image. The extraction process becomes more difficult when images are transmitted over a network. Many intentional and unintentional alterations can be applied on it. In such cases, survival of watermark image is desirable. A watermark that survives different attacks is called a robust watermark. Another requirement of the watermark is the invisibility, which should keep the perceptual quality of the original images high. The watermark is preferred to be totally invisible. Robustness and invisibility are contrary, i.e. increasing the robustness means embedding the watermark with more intensity which decrease the quality of the watermarked image. On the other hand, using low embedding intensity will produce higher quality watermarked images but with less robustness. Watermarking can be implemented as time domain or transform domain processes. Time domain attempts are characterized by simplicity and low computation overhead, this study implemented watermarking by using the spatial domain [3].

1.2 literature review

In 2008, Houtan Haddad Larijani, and Gholamali Rezai Rad; proposed a new algorithm for watermarking in spatial domain. They choosed spatial domain to take advantage of its relatively low calculation complexity when compared to any technique requiring domain transforms. The algorithm was implemented in pixel by pixel comparison between host image and watermark pixels. For each pixel of the host image, if its corresponding value equals to the compared pixel in the watermark, then save its position as a key. Therefore, they called this algorithm as "Save Algorithm" [4].

In 2018, Ghadi, L. Laouamer, L. Nana, and A. Pascu; proposed a study of a watermark embedding using the spatial domain depending on the analysis of texture and associated rules were performed. The researchers suggested to embed a watermark in highly textured places in the host image in order to increase and enhance the imperceptibility and the robustness [5].

In 2019, Qingtang Su, Decheng Liu, et-al; proposed a novel spatial domain color image watermarking technique to rapidly and effectively protect the copyright of the color image. First, the direct current (DC) coefficient of 2D-DFT obtained in the spatial domain was discussed, and the relationship between the change of each pixel in the spatial domain and the change of the DC coefficient in the Fourier transform was proved. Then, the DC coefficient was used to embed and extract a watermark in the spatial domain by the proposed quantization technique[6].

In 2020, Sanjay Kumar1 & Binod Kumar Singh;Here digital image watermarking technique based on LSB Substitution and Hill Cipher presented and examined in this paper. For better imperceptibility a watermark was inserted in the spatial domain. Further the watermark was implanted in the Cover Image block having the highest entropy value. To improve the security of the watermark hill cipher encryption was used. Both subjective and objective image quality assessment technique was used to evaluate the imperceptibility of the proposed scheme [7].

2. Theoretical Background

Digital watermarking includes two major processes: embedding and extracting. The watermark is inserted into the host data during the embedding procedure. After embedding the watermark, the original digital data (multimedia content) will be somewhat changed, and this changed data is referred to as watermarked data. During the extraction procedure, the embedded watermark is removed from the watermarked data. The retrieved watermark is then matched with the original watermark; if the watermarks are identical, the data is considered as authenticated. [8]

Figure (1) shows a typical watermarking system, which comprises a watermark embedder and a watermark extractor.[9] [10].

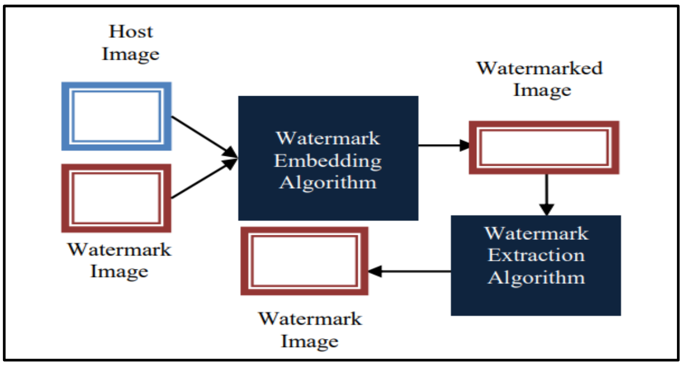


Figure 1 : A Typical Watermarking System [10]

2.1 Types of Watermarking Systems

The following are several kinds of watermarking methods:

a- **Robust and Fragile Watermarking:**In robust watermarking, any modifications to the watermarked image or video will not affect the watermark, while in fragile watermarking, any changes to the watermarked material will alter or destroy the watermark.[11]

b- **Visible and Invisible Watermarking:** Visible watermarking occurs when the material is visible to the human eye, while transparent watermarks, also known as invisible watermarks, occur when the content is not visible to the human eye.[12]

c- **Blind and Non-blind Watermarking:** The techniques of watermarking are commonly classified into two prominent types: non-blind or blind, it depends on necessity of the host image to extract the watermark image. As it is known, the algorithms of non-blind watermarking are inconvenient for the practical applications as they need the non-watermarked image to be presented throughout the extraction stage.[13]

d- **Spatial Domain andFrequency Domain Watermarking:** Watermarking algorithms may be based on a spatial domain or a transformed domain. The pixel values of one or more randomly chosen subsets of pictures are altered via spatial domain watermarking. It puts the raw data straight into the image pixels. It may be based on the use of various patches or least important bit planes. The change of transform coefficients is the basis for transform domain watermarking. DCT and DWT are two of the most frequently utilized transform domain techniques.[14]

2.2 Attacks

Different types of attacks occur on the watermarked media when it is transmitted. This research tackles some of those attacks, which might be manifested as follows [15]:

Interference Attacks: Concerning this sorts of attacks it inserts noise to the watermarked media, and there are several examples for this type, including quantization, compression, etc.

Geometric Attacks: In this type geometry of the image ischanged. Examples of this type: cropping, rotation, etc.

Active Attacks: These attacksare considered as the most important ones. The user who is not authorized attempts to detect the watermark or creates another watermark so that detection cannot be performed disregarding the operation used.

Image DegradationAttacks: In attacks of this type, parts of the image are removed, this results in watermarks partial or fulldamage. Examples of these attacks are: partial cropping and Gaussian noise.

2.3 Peak Signal to Noise Ratio :

The watermarked images quality can be evaluated by using, some parameters like:" MSE and PSNR ".
"(MSE) Mean Squared Error": this is resulted of the squared average of the difference between the host image and the watermarked one. It is calculated by using equation (1):

$$MSE = \frac{1}{PQ}\left(\sum_{i=1}^{P}\sum_{j=1}^{Q}\left(m(i,j) - n(i,j)\right)\hat{}2\right) \qquad (1)$$

Where; P and Q are the image "height and width" respectively. Moreover, m(i,j) stands for the pixel's value in the host image and n(i,j) stands for the pixel's value of the embedded image [11].

(PSNR) Peak Signal to Noise Ratio : It determines the ratio of quality in the watermarked image in relation to the host image as in equation (2):

$$PSNR = 10log_{10}\left(\frac{L*L}{MSE}\right) \qquad (2)$$

Where: L is the highest pixel value of the image, i.e. for image of "8" bits, L=255 [16].

2.4 Normalization Cross Correlation(N.C.)

The watermark will be exposed to a variety of well-known attacks, and the scheme's efficacy will be evaluated. A normalized correlation, which is the correlation between the original and recovered image, is computed to measure the picture's robustness, or how well it can resist an assault, as described by the following equation:

$$Nc(W,W') = \frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} W'(i,j)}{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[W(i,j)]^2} (3)$$

Where (W), and (W') are the embedded and the extracted watermarks respectively, each with dimensions M*N [17]. With NC =1, meaning no change.

2.5 Raspberry Pi 4

Raspberry Pi is a card machine developed in February 2012 by the Raspberry Pi Foundation, which can execute small to medium tasks that can be done by a conventional desktop computer. It provides major improvements in processor speed, multimedia performance, memory, and connectivity compared with the prior-version “Raspberry Pi 4 Model B” while maintaining drawbacks as the consumption of power. [18]. As seen in Figure (2)

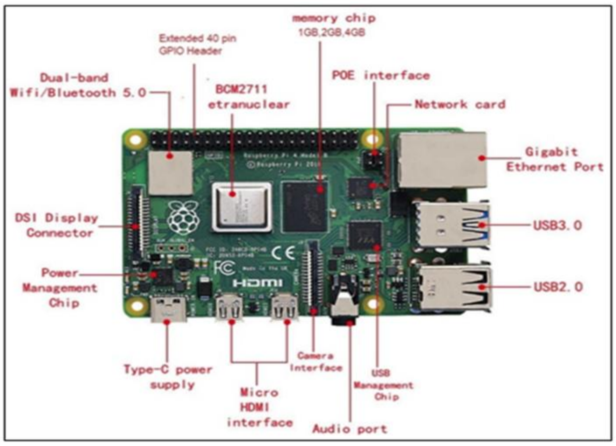


Figure 2 : Raspberry Pi4 Model

3. Methodology

In order to complete the requirements of concealment, durability and strength, the study system was proposed, which includes dividing the host image into a number of blocks equal to the number of pixels of the watermark, the dimensions of each block are inversely proportional to the dimensions of the watermark. A symmetric key was used to encrypt the watermark before burial and re-decrypt it after retrieval to increase security in the two aforementioned operations.

3.1 Watermark Image Modulation

Let The watermark image, I=[i_1, i_2, i_3, … i_n] with $i_i \in$ {0,1} is an n-bit sequence of any watermark image like a name or logo for the owner.

The watermark image is encrypted or modulated via bitwise XOR operation with a secret key bit sequence K=[k_1, k_2, k_3, … k_n] with $k_i \in$ {0,1} to obtain the modulated watermark image W=[w_1, w_2, w_3, … w_n] with the same size of I and K, as shown in Figure (3).

Seed number is used as a pseudo-random number generator is regarded as a private key for the proposed algorithm for embedding and extraction process.

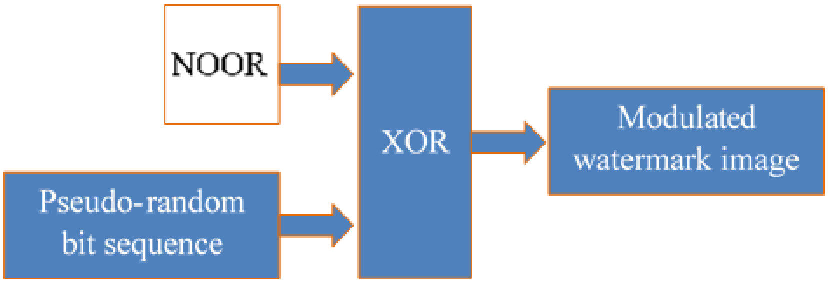


Figure 3 : The process of watermark image modulation

3.2 Embedding Algorithm

Before embedding process the original host image is splinted into (n*n) blocks which equals to the watermark image dimension. The dimension of each block is calculated as follows:

$n = ((M*N)/(X*Y))^{1/2}$

Where M and N are the dimensions of the host image, X and Y are the dimensions of the watermark image.

The embedding process in each block of the host image is processed according to the steps below:

1. Read the original host image.
2. Read the watermarkimage.
3. Subdivide the host image into (n*n) blocks which equals to the watermark image dimensions.
4. Read seed number for encrypting the watermark image
5. Encrypt (Modulate) the watermark image using binary random sequence numbers depending on the seed number via XOR operation.
6. Get the first pixel (bit) of the watermark image.
7. If the pixel value of the watermark image = 0 then clear (b1=0) of all 8-bit block pixels, otherwise set (b1=1), where b1 is the bit next to least significant bit (with weight=2^1) of each 8-bit pixel in that block.
8. If there is an additional pixel in the watermark image, get the next pixel of the watermark image and go to step 7, otherwise;
9. End.

Figure (4) shows flow chart of the embedding process.

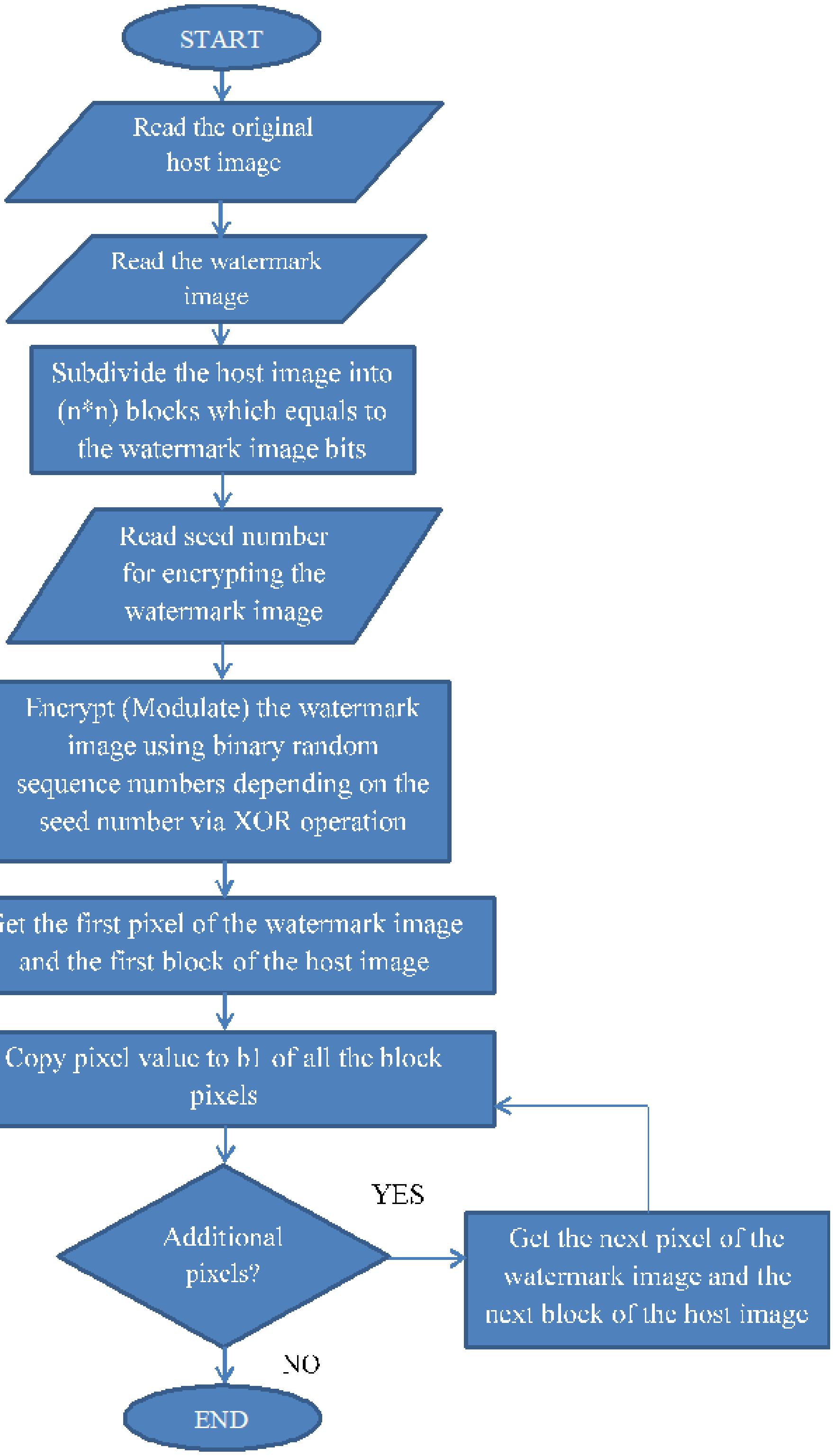

START
Read the original host image
Read the watermark image
Subdivide the host image into (n*n) blocks which equals to the watermark image bits
Read seed number for encrypting the watermark image
Encrypt (Modulate) the watermark image using binary random sequence numbers depending on the seed number via XOR operation
Get the first pixel of the watermark image and the first block of the host image
Copy pixel value to b1 of all the block pixels
Additional pixels?
YES
Get the next pixel of the watermark image and the next block of the host image
NO
END

Figure 4 : flowchart of the embedding process

3.3 Extraction Algorithm

According to the procedure of embedding, the process of extraction is done as the following:

1. Read the watermarked image.
2. Read the seed number to decrypt the watermark image
3. Subdivide the watermarked image into n*n blocks.
4. Locate the current block of the watermarked image.
5. Get the value of b1 of each pixel in the current block, and then calculate the number of zeros and the number of ones in each block.
6. If the number of zeros is greater than the number of ones in the current block, then the current pixels of watermark image is “0” otherwise the current pixel is “1”.
7. If there are additional blocks in the watermarked image, get the next block of the watermarked image, go to step 4.
8. Decrypt the watermark image by using the binary random sequence numbers depending on the seed number.
9. Display the watermark image.
10. End.

Note: The N.C. is nearest to 1 when the size of the host image is much greater than the watermark image size because the size of all blocks will be bigger.

Figure (5) shows flowchart of the extraction process.

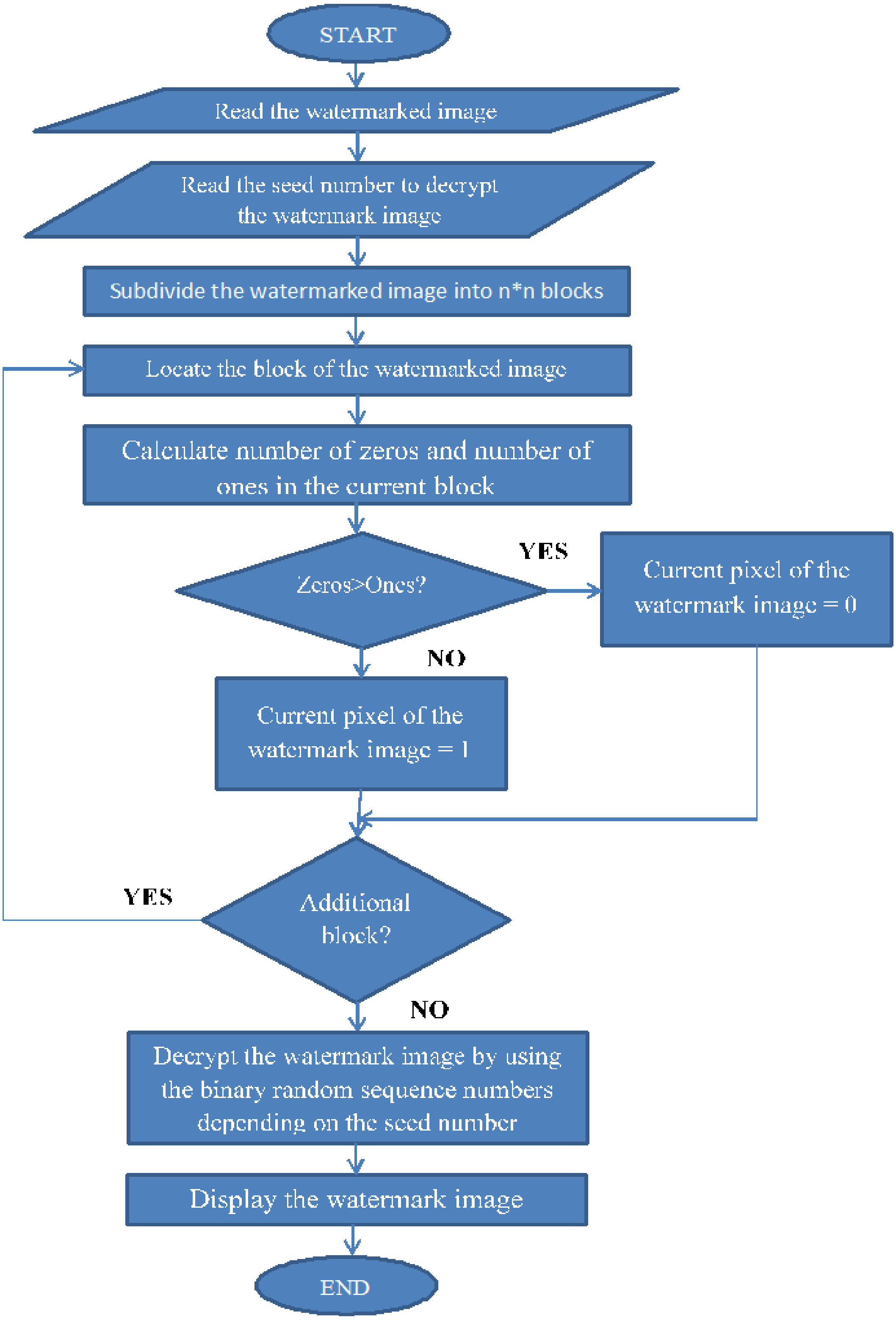

START
Read the watermarked image
Read the seed number to decrypt the watermark image
Subdivide the watermarked image into n*n blocks
Locate the block of the watermarked image
Calculate number of zeros and number of ones in the current block
Zeros>Ones?
YES
Current pixel of the watermark image = 0
NO
Current pixel of the watermark image = 1
YES
Additional block?
NO
Decrypt the watermark image by using the binary random sequence numbers depending on the seed number
Display the watermark image
END

Figure 5 : flowchart of the extraction process

3.4Implementation on Raspberry pi4

PYTHON, was used to implement the above algorithm under UBUNTO environment. The Raspberry Pi4, its specifications are (Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz, 4G RAM, SD card 64G).[18]

The main important libraries used in proposed system implemented in PYTHON are : [19]

- **Numpy (Numerical Python):** It provides a high-performance multidimensional array object and tools for working with these arrays.
- **Tkinter (Tk interface):** The tkinter package is the standard Python interface to the Tk GUI toolkit.
- **Pillow:**This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.

Figure (6) illustrates a brief block diagram of the implementation above proposed system on Raspberry Pi4.

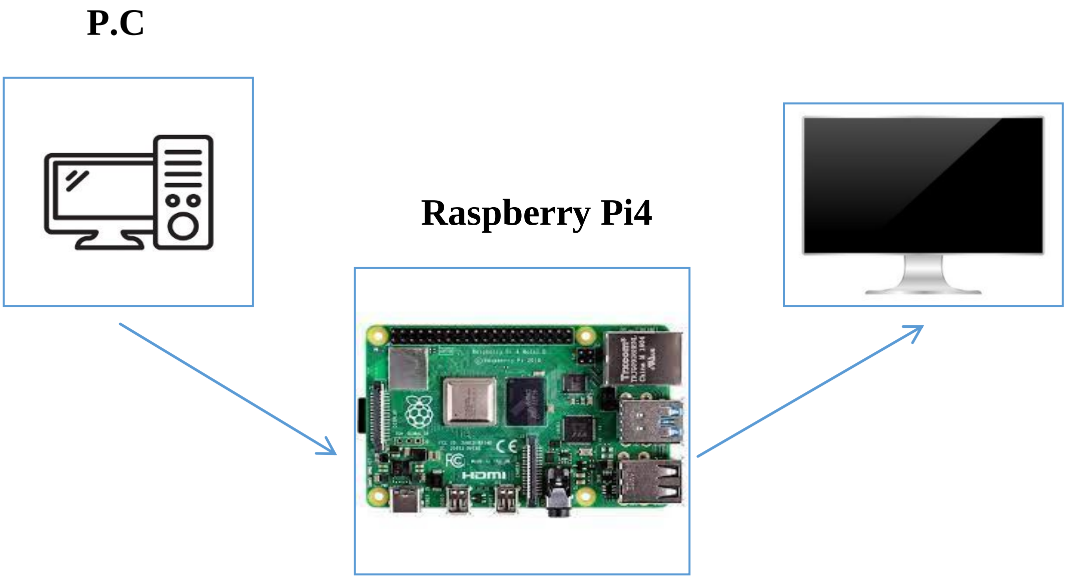
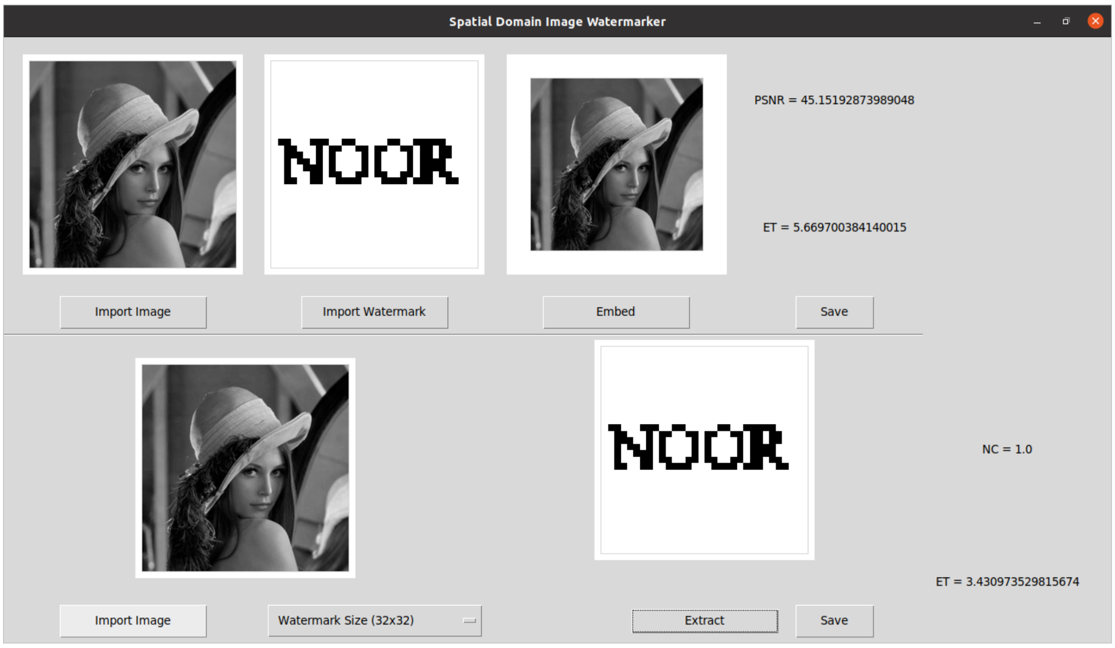


Figure6 :Block diagram of implementation of the proposed system on Raspberry Pi4

4. Experimental Results

The proposed algorithm has been evaluated by using the standard Lena and Bird images, with image dimensions of 512*512 pixels as the host image where the binary logo image (NOOR) was chosen to be of 32*32 bits

The results show that the watermarked image quality is good and as will be shown it is robust to many attacks.



PSNR = 45.15192873989048

ET = 5.669700384140015

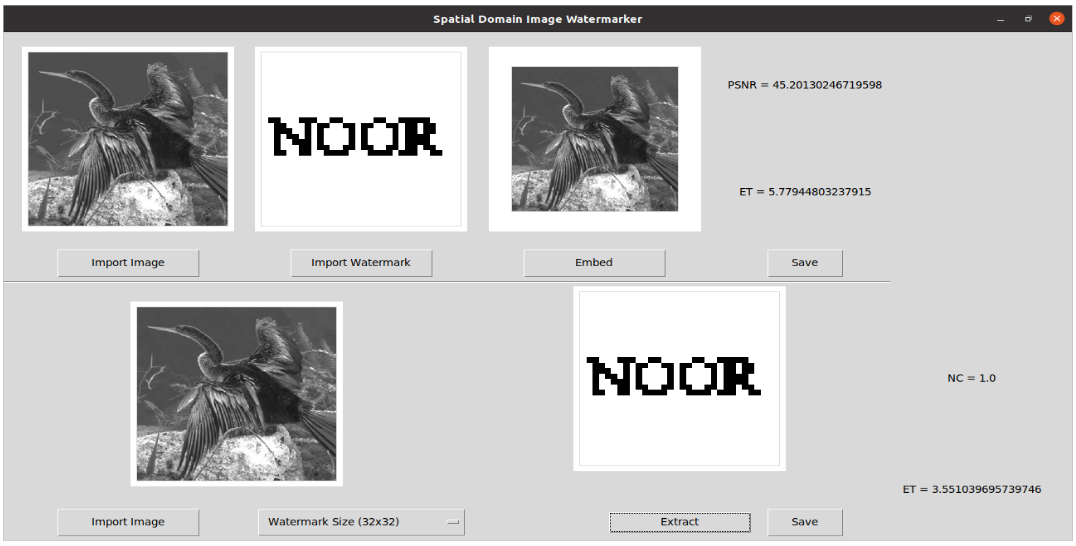
NC = 1.0

ET = 3.430973529815674

Figure 7: Embedding and extracting the watermark in Lena image.

Figure (7) shows the hiding process of (NOOR) watermark image in the (Lena) image, where the value of PSNR = 45.151 dB, which is an acceptable value. From the other hand, the same figure shows the watermark extracting process, and the value of NC = 1 because no attack was launched on the image.

Figure (8)demonstrates that thesame watermark is also hidden in the Bird image, the value of PSNR = 45.201 dB which is also acceptable.In the same Figure, it displays the result of extracting the watermark image at the value of NC = 1.



PSNR = 45.20130246719598

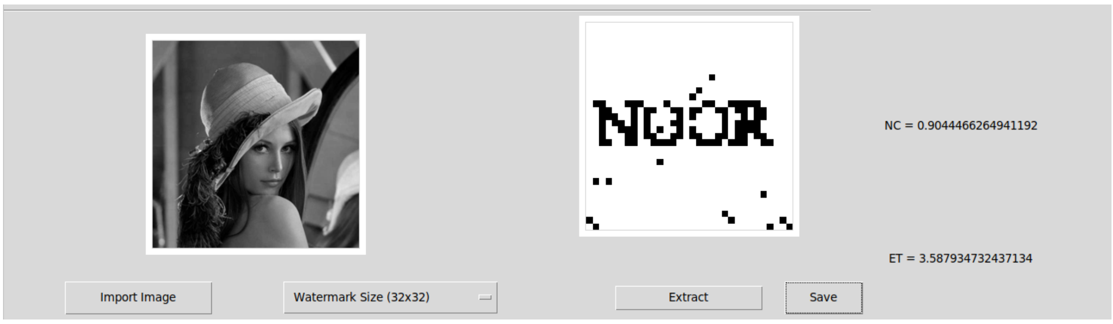
ET = 5.77944803237915

NC = 1.0

ET = 3.551039695739746

Figure 8 : Embedding and extracting the watermark in Bird image.

Figure (9) manifests the process of extracting the watermark image after determining its size and the value of NC = 0.9044 after attacking the image by compressing it with a value of 85%.

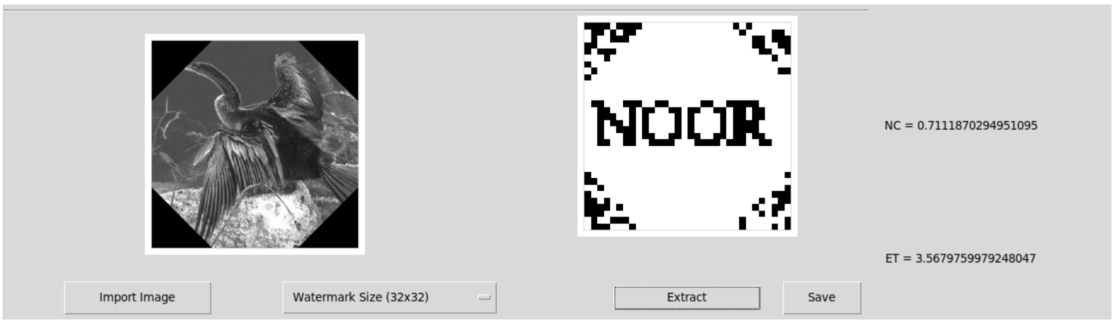


NC = 0.9044466264941192

ET = 3.587934732437134

Figure 9: Extracting watermark from Lena image after a compression attack

Figure (10) displays the process of extracting the watermark image՝ here the value of NC = 0.7111 after attacking the imageby rotating it by 45 degrees in the clockwise direction.



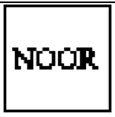
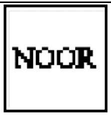
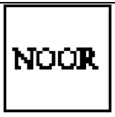
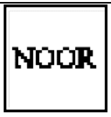
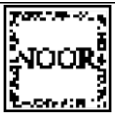
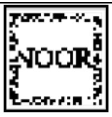
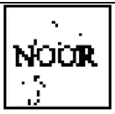
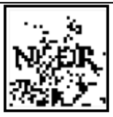
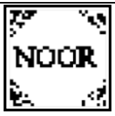
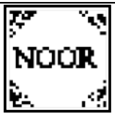
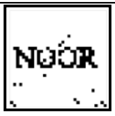
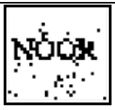
NC = 0.7111870294951095

ET = 3.5679759979248047

Figure 10: Extracting watermark from Bird image after a rotation attack

Table (1) demonstrates the overall processes of embedding and extracting for the (NOOR) watermark image on two host images (Lena and Bird) with five types of attacks and the results were very acceptable.

Table 1: The overall processes of embedding and extractingresults

Attack/Image	**Lena** **PSNR=45.151**	**Bird** **PSNR=45.201**
No attack	 N.C. =1	 N.C. =1
Median filter	 N.C. =1	 N.C. =1
Cropping (20%)	 N.C. =0.5299	 N.C. =0.5299
Rescaling (20%)	 N.C.=0.887	 N.C.=0.4071
Rotation (45) degree	 N.C.=0.7111	 N.C.=0.7111
Compression (85%)	 N.C.=0.9044	 N.C.=0.801

5. Conclusions and Future Works

Through working within the subject of digital watermarking in this research, the following conclusions have been obtained. They are described in the points listed below.

•The increase in exchanging data made it necessary to secure those data in what is known to be Information Security, therefore the concept of watermark emerged to ensure the copyrights of multimedia.

•Choosing the block based embedding was done to increase the robustness, were each bit is embedded repeatedly for a whole block, so giving better chances to retrieve it correctly after the attacks.

• embedding the watermark has to be in the bits of data that are the least significant to ensure a minimum effect on the cover image and an increase in the PSNR value as embedding and extraction is robust and the NC value gets closer to “1”.

•This embedding procedure showed very good robustness for attacks that does not change the pixels values in the chosen sub images. These attacks were median filtering, cropping and rotation.

- For future work It could be useful to use the newly established FPGA kits which have more facilities and memory. This could be important to get a single chip stand-alone system if it were needed.

References

[1] Deepshikha Chopra, Preeti Gupta, *et-al*, "Lsb Based Digital Image Watermarking For Gray Scale Image", IOSR Journal of Computer Engineering (IOSRJCE) ISSN: 2278-0661, ISBN: 2278-8727 Volume 6, Issue 1 (Sep-Oct. 2012), PP 36-41, 2012

[2] B Surekha, Dr GN Swamy, "A Spatial Domain Public Image Watermarking", International Journal of Security and Its Applications, Vol. 5 No. 1, January, 2011

[3] Rafael C Gonzalez, Richard E. Wood, "Digital Image Processing", Fourth Edition, 2018

[4] Houtan Haddad Larijani, Gholamali Rezai Rad, "A New Spatial Domain Algorithm for Gray Scale Images Watermarking", Proceedings of the International Conference on Computer and Communication Engineering 2008 May 13-15, 2008 Kuala Lumpur, Malaysia, 2008

[5] Musab Ghadi, Lamri Laouamer, Laurent Nana, Anca Pascu, "A blind spatial domain-based image watermarking using texture analysis and association rules mining", Received: 5 March 2018 /Revised: 19 September 2018 /Accepted: 6 November 2018, © Springer Science+Business Media, LLC, part of Springer Nature, 2018

[6] QINGTANG SU, DECHENG LIU, et-al, "New Rapid and Robust Color Image Watermarking Technique in Spatial Domain", 2169-3536 2019 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission.http://www.ieee.org/publications_standards/publications/rights/index.html for more information, 2019

[7] Sanjay Kumar, Binod Kumar Singh, "Entropy based spatial domain image watermarking and its performance analysis", Multimedia Tools and Applications https://doi.org/10.1007/s11042-020-09943-x, 2019

[8] F. Sebe, T. Domingo-Ferrer, J. Herrera, "Spatial Domain Image Watermarking Robust against Compression, Filtering, Cropping and Scaling", Springer Verlog, LNCS, 1975, pp. 44-53, 2000

[9] I. J. Cox, M. L. Miller, J. A. Bloom, "Digital Image Watermarking", Morgan Kaufman, Publishers, USA, 2004

[10] Jasmine Selvakumari and Suganthi Jeyaraj, "Using Visible and Invisible Watermarking Algorithms for Indexing Medical Images", *The International Arab Journal of Information Technology, Vol. 15, No. 4, July 2018*

[11] D. Zheng, Y. Liu, J. Zhao, A. Saddik, "A Survey of RST Invariant Image Water-marking Algorithms", ACM Computing Surveys Vol. 39, No. 2, Article 5, 91 pages, 2007

[12] *Rowayda, A. Sadek , "Blind Synthesis Attack on SVD Based Watermarking Techniques". 2008 International Conference on Computational Intelligence for Modeling Control & Automation, Vienna, Austria: 140–145. doi:10.1109/CIMCA.2008.53.*

[13] Mohammad Ghebleh, Ali Kanso and Hala S. Own, "A blind chaos-based watermarking technique", Published online 30 April 2014 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.783

[14] M. Barni, F. Bartolini, "Watermarking Systems Engineering", Signal and Commmu-nication

Series, Marcel Dekker Inc. USA, 2004

[15] Amit Kumar Singh, Nomit Sharma, Mayank Dave, Anand Mohan, 2012, "A Novel Technique for Digital Image Watermarking in Spatial Domain" , 2012 2nd IEEE International Conference on Parallel, Distributed and Grid Computing.

[16] Manish Gupta, "Optimized Digital Image Watermarking for Uncorrelated Color Space", RAJASTHAN TECHNICAL UNIVERSITY, KOTA, 2015, ALL RIGHTS RESERVED

[17] Kiran Sultan, Dhiaa Musleh, *et-al*, "Robust and Fragile Medical Image Watermarking: A Joint Venture of Coding and Chaos Theories", Journal of Healthcare Engineering / 2018 / Article

[18] https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-4-Product-Brief.pdf

[19] Richard Blum, "Python Programming for Raspberry Pi in 24 Hours", Copyright © 2014 by Pearson Education, Inc.