Image Watermarking using least significant bit algorithm

[1]FENWA O.D, [2]AJALA F.A , [3] ALO O.O

[1,2,3] Ladoke akintola university of technology, pmb 4000, ogbomoso

Abstract

The advent of the Internet has resulted in many new opportunities for the creation and delivery of content in digital form. Applications include electronic advertising, real-time video and audio delivery, digital repositories and libraries, and Web publishing. An important issue that arises in these applications is the protection of the rights of all participants. It has been recognized for quite some time that current copyright laws are inadequate for dealing with digital data. This has led to an interest towards developing new copy deterrence and protection mechanisms. In this paper, a digital watermarking system was developed using Least Significant Bit (LSB) algorithm. C# programming language was used to implement the system and its robustness was tested using different types of attacks such as distortion, saturation and gray scale.

Keywords: Image watermarking, copyright protection, secret message, digital image,

1. Introduction

In recent years, the distribution of works of art, including pictures, music, video and textual documents, has become easier. With the widespread and increasing use of the Internet, digital forms of these media (still images, audio, video, text) are easily accessible. This is clearly advantageous, in that it is easier to market and sell one's works of art. However, this same property threatens copyright protection. Along with the advancement of Image technologies in the past decades, storage of data by digital products or transmission of images over the Internet, in the form of texts, images or videos, have brought about significant progress for modern information technology. Nonetheless, the increasingly powerful software has also made it easy to gain unrestricted access to the data in the storage media for further modifying its content. As a result, data and copyright protection have been important subjects in research and applications nowadays. On the other hand, during the process of data transmission, it often occurs that confidential information needs to be encrypted in order to avoid interception by attackers in communication networks. To solve these problems, several information hiding methods have been proposed and studied for data protection and secret information embedding, but these traditional data hiding techniques often bring about permanent damage to the content of the host media, which is not acceptable in some applications [1].

Digital documents are easy to copy and distribute, allowing for pirating. There are a number of methods for protecting ownership. One of these is known as digital watermarking. Watermarks of varying degrees of obtrusiveness are added to presentation media as a guarantee of authenticity, quality, ownership, and source. To be effective in its purpose, a watermark should adhere to a few requirements. In particular, it should be robust, and transparent. There two major types of watermarking techniques, which are visible and invisible watermarking techniques, Visible watermarks, as the name says, are visual patterns, like logos, which are inserted into or overlaid on images (or video), very similar to visible paper watermarks can be seen clearly by the viewer and can also identify the logo or the owner. Visible watermarking technique changes the original signal. The watermarked signal is different from the original signal. Visible watermark embedding algorithms are less computationally complex. The watermarked image cannot withstand the signal processing attacks, like the watermark can be cropped from the watermarked image [2].

Unlike visible watermarking, invisible watermarks cannot be seen by the viewer. The output signal does not change much when compared to the original signal. The watermarked signal is almost similar to the original signal. As the watermark is invisible, the imposter cannot crop the watermark as in visible watermarking. Invisible watermarking is more robust to signal processing attacks when compared to visible watermarking. As the quality of the image does not suffer much, it can be used in almost all the applications. A digital image version of an analogue image contains sampled values of the function at discrete locations or pixels. These values are said to be the representation of the image in the spatial domain or often referred to as the pixel domain. Spatial embedding inserts message into image pixels. The oldest and the most common used method in this category is the insertion of the watermark into the Least Significant Bits (LSB) of pixel data [3], [4] and [5]. LSB coding is one of the earliest methods. It can be applied to any form of watermarking.

In this method, the LSB of the carrier signal is substituted with the watermark. The bits are embedded in a sequence which acts as the key. In order to retrieve it back this sequence should be known. The watermark encoder first selects a subset of pixel values on which the watermark has to be embedded. It then embeds the information on the LSBs of the pixels from this subset. In computing, the least significant bit (LSB) is the bit position in a binary integer giving the units value, that is, determining whether the number is even or odd. The LSB is sometimes referred to as the right-most bit, due to the convention in positional notation of writing less significant digit further to the right. It is analogous to the least significant digit of a decimal integer, which is the digit in the ones (right-most) position.

2. Material and Method

Stages of the system development

- ❖ Image Acquisition/Selection
- ❖ Development of an invisible digital watermarking software using LSB algorithm
- ❖ Encoding process
- ❖ Decoding process
- ❖ Implementation of an image watermarking system using the C-sharp programming language
- ❖ Evaluate of performance of the image watermarking system using different types of attach such as distortion, gray scale, saturation,

2.1 Image Acquisition

Images were acquired through various media including friends' photo gallery on mobile phones and also from the internet. The image format used was bitmap.

2.2 The Encoding Process

The most commonly used method to embed a bit is Least Significant Bit (LSB) embedding as shown in figure 2.1, where the least significant bit of a Bitmap (BMP) coefficient is modified in order to embed one bit of message. Once the required message bits have been embedded, the modified coefficients are compressed using entropy encoding to finally produce the BMP watermark image. By embedding information in BMP coefficients, it is difficult to detect the presence of any hidden data since the changes are usually not visible to the human eye in the spatial domain. The embedding process of the LSB technique can be illustrated as follows: Consider that the system is required to hide a watermark number 178 in a 2x2 gray-scale (8-bit) image. Let's assume that the image pixels are 234, 222, 190 and 34. In an 8-bit binary format, the number 178 is represented as 10110010. Since there are 4 pixels that can be used to store this data, we can easily decide to embed pairs of bits of the watermark to the last 2 insignificant bits of the pixels. The process therefore modifies the original bits from 11101010, 11011110, 10111110 and 00100010 to 11101010, 11011111, 10111100 and 00100010 respectively [6].

2.3 The Decoding Process

During the extraction process, the BMP file is entropy decoded to obtain the BMP coefficients, from which the message bits are extracted from the LSB of each coefficient as shown in figure 2.2. After due considerations, the Least Significant Bit Algorithm was used to encode the texts on bitmap images using the C-sharp programming language to code.

2.4 Least Significant Bit Algorithm

- A raw bitmap image **'C'** is selected from the set of standard test images. Let this be the base image on which the watermark will be added.
- Divide the image into four by four segments.
- Extract the binary values of **C**.
- Input the Secret Message, **M**. This will be the watermark which will be added to the base image
- Encode the **M** in binary.
- Read the least significant bit LSB of **C**
- Use a pixel selection to hide information of **M** in the LSB of **C**
- Save the new image (watermarked image-object) **S**

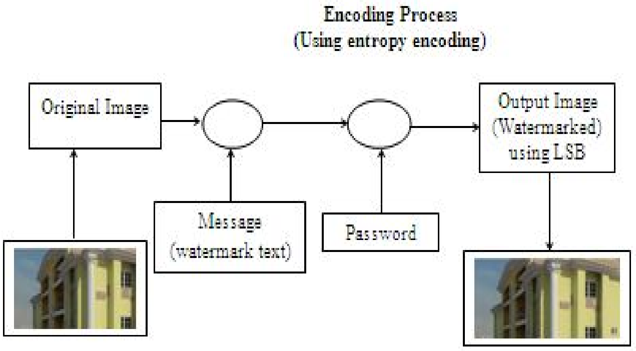


Figure 2.1: Image watermarking (embedding hidden message)

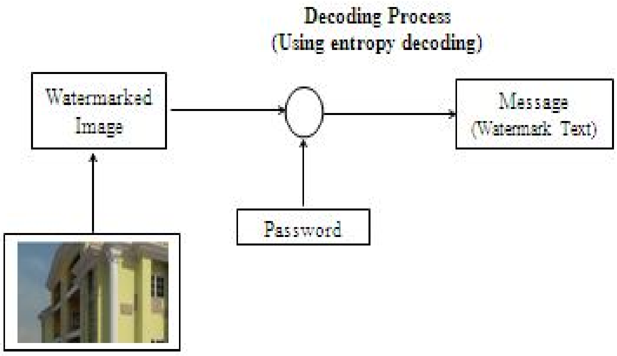


Figure 2.2: Image Verification (Owner's assertion)

3. Implementation

The system was developed using C#, the user interface as shown in figure 3.1 is used to load the image into the system and then further processing of the image using digital watermarking is carried out.

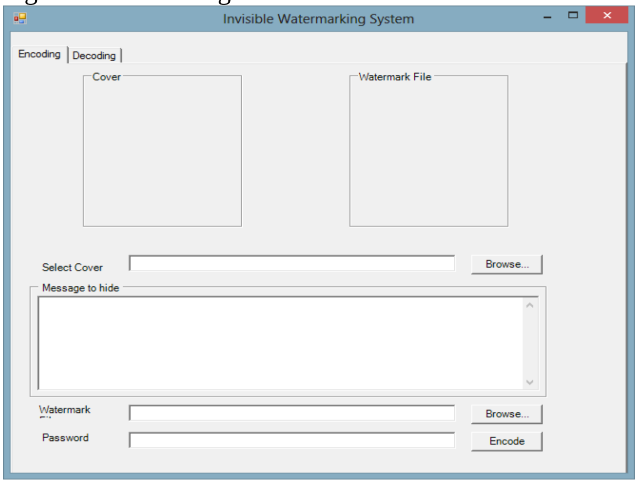
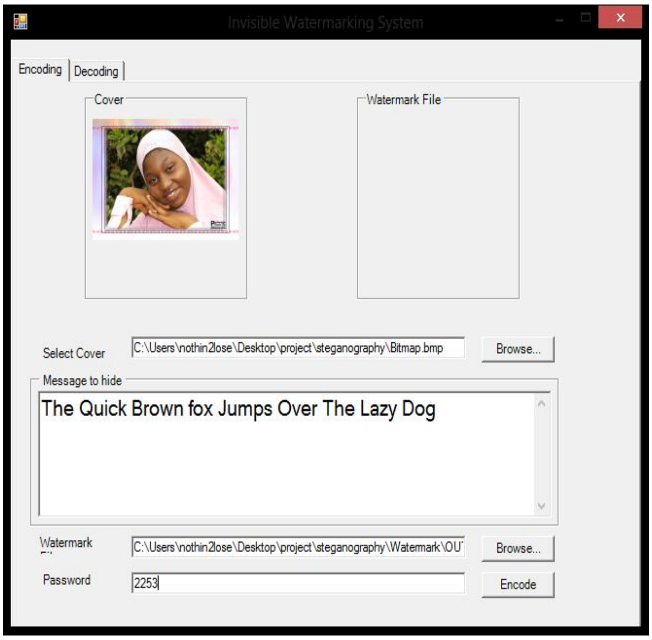


Figure 3.1: The homepage of the developed watermarking system

The homepage has two tabs; Encoding and Decoding. The encoding tab is selected if the user intends to watermark a particular image. This is where a hidden message (watermark) is added to a selected image. Hence, the encoding option is chosen.



Select Cover: C:\Users\nothin2lose\Desktop\project\steganography\Bitmap.bmp

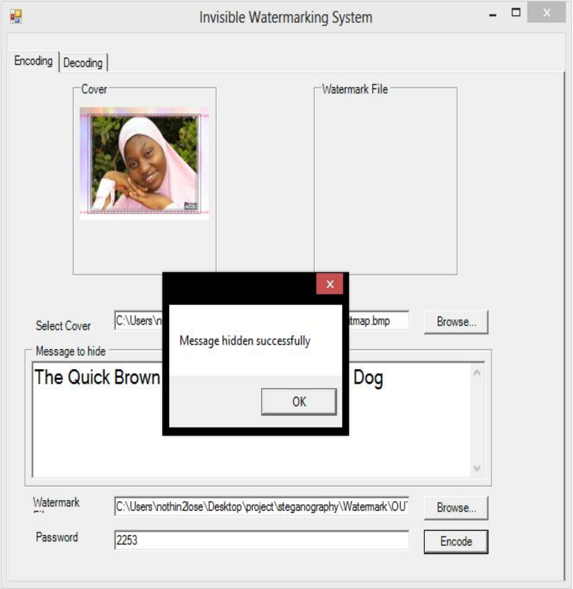
Message to hide: The Quick Brown fox Jumps Over The Lazy Dog

Watermark: C:\Users\nothin2lose\Desktop\project\steganography\Watermark\OU'

Password: 2253

Figure 3.2 shows the software with hidden text, selected target image and password included in the entry field above.

In figure 3.2, a cover image (that is, the image to be watermarked) is selected and the message to hide (watermark) is supplied as indicated in figure 3.2. A name and storage location for the watermarked image is also specified and the also is the password. The password will be required to decode the watermarked image so as to reveal the hidden message. After supplying the required information, the encode button to hide the text into the image is selected.



Message hidden successfully

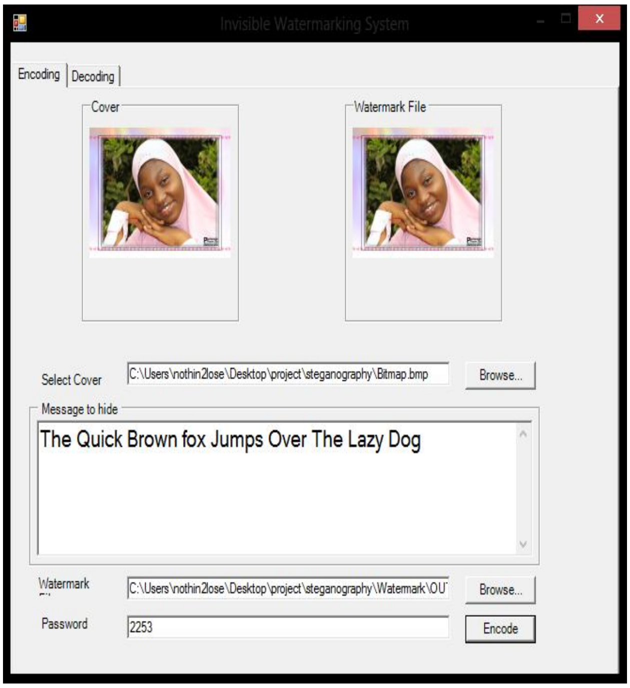
Select Cover: C:\Users\n ... itmap.bmp

Message to hide: The Quick Brown ... Dog

Watermark: C:\Users\nothin2lose\Desktop\project\steganography\Watermark\OU'

Password: 2253

Figure 3.3: The interface with a dialog box confirming the hidden message has been embedded above.



Select Cover: C:\Users\nothin2lose\Desktop\project\steganography\Bitmap.bmp

Message to hide: The Quick Brown fox Jumps Over The Lazy Dog

Watermark: C:\Users\nothin2lose\Desktop\project\steganography\Watermark\OU'

Password: 2253

Figure 3.4: The interface with the watermarked image on the right.

Figure 3.4 above shows the cover image on the left and the watermarked image on the right. This shows there's no visible difference between the watermarked image and the original image.

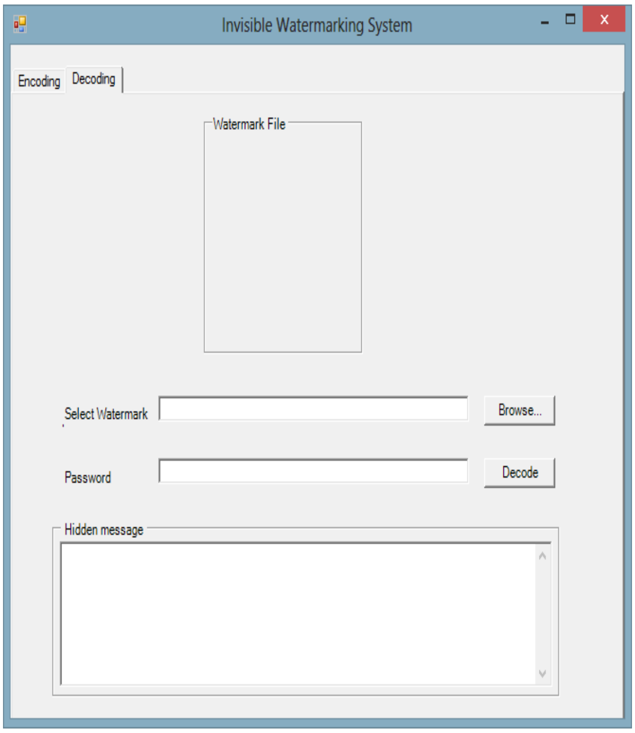
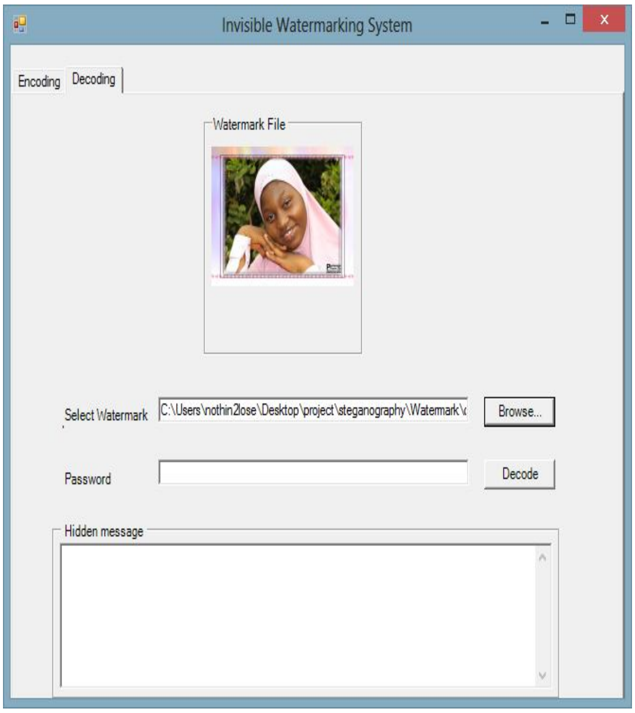


Figure 3.5: The homepage of the decoding process

Figure 3.5 is the homepage of the decoding process chosen to decode (reveal the watermark text) in an already watermarked image.

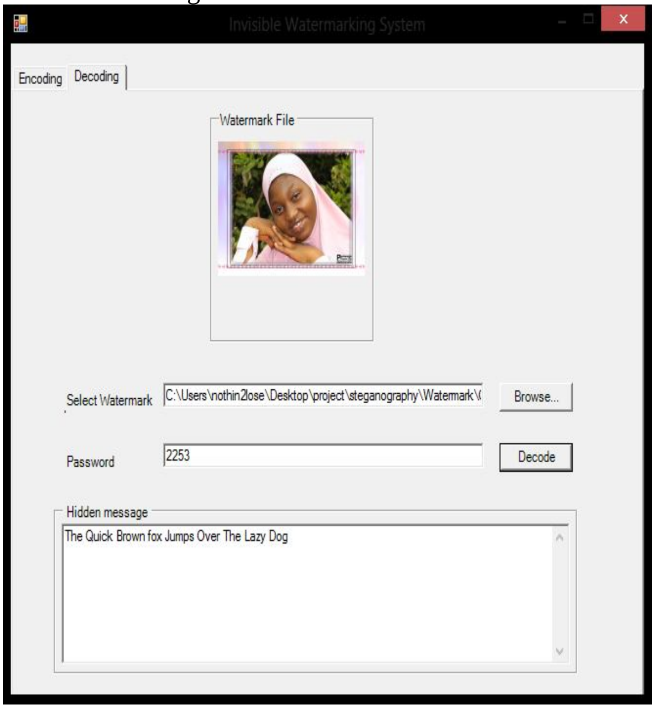


Select Watermark: C:\Users\nothin2lose\Desktop\project\steganography\Watermark\

Password:

Figure 3.6: Interface showing how the watermarked image is selected in order to retrieve the embedded message.

In figure 3.6 the watermarked image to be decoded has been chosen using the "select watermark" button.



Select Watermark: C:\Users\nothin2lose\Desktop\project\steganography\Watermark\

Password: 2253

Hidden message: The Quick Brown fox Jumps Over The Lazy Dog

Figure 3.7: The interface in which the correct password has been entered and the hidden message has been retrieved.

In figure 3.7, the password is entered and the decode button was clicked which makes the software reveal the text embedded in the image during the encoding stage.

4. Performance Evaluation

Different attacks that are applied to the watermarked image to test the robustness of watermarking system are (i) distortion, (ii) saturation and (iii) gray scale. The extracted watermarks after applying various attacks are shown in figure 4.1 to 4.6.

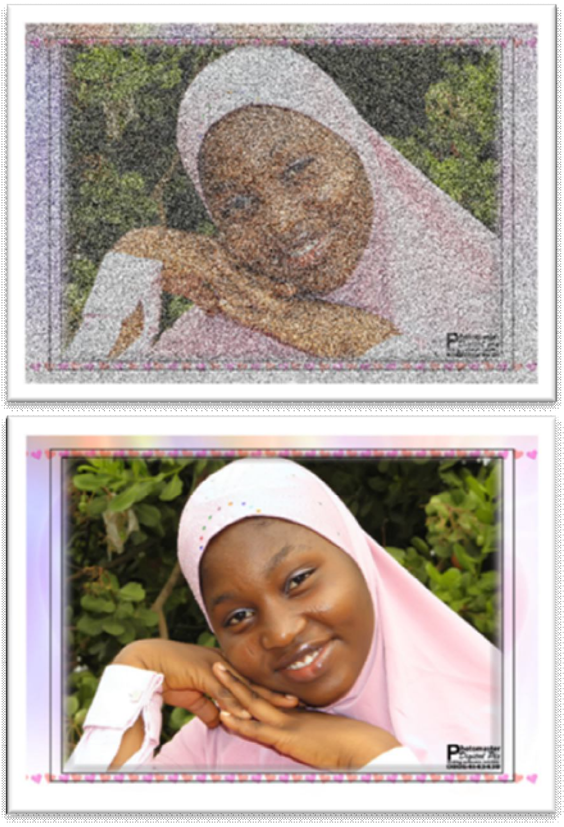
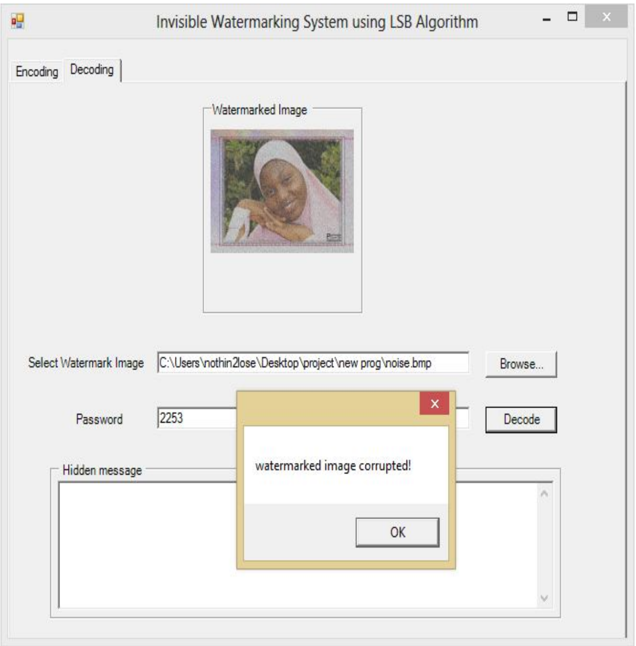


Figure 4.1: Watermarked image with no attack on the left and watermarked image with noise distortion on the right.



Select Watermark Image: C:\Users\nothin2lose\Desktop\project\new prog\noise.bmp

Password: 2253

watermarked image corrupted!

Figure 4.2: The interface showing error message when the watermarked image is attacked with noise distortion was decoded.

In figure 4.2, the system displays the "watermarked image corrupted" on trying to decode a watermarked image that has been attacked with noise.

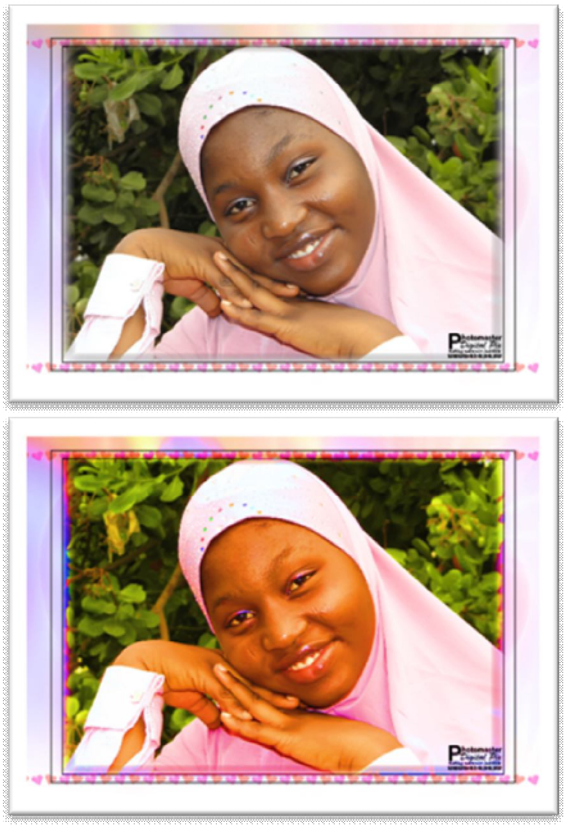
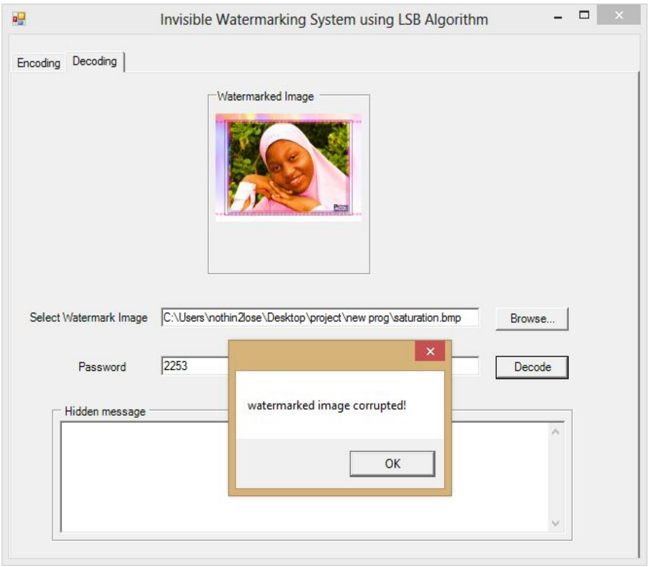


Figure 4.3 Watermarked image with no attack on the left and watermarked image with saturation attack on the right.



Select Watermark Image C:\Users\nothin2lose\Desktop\project\new prog\saturation.bmp

Password 2253

watermarked image corrupted!

Figure 4.4: Interface displaying error message when the watermarked image attacked with saturation was to be decoded.

Figure 4.4 displays the "watermarked image corrupted" on trying to decode a watermarked image that has been saturated using image editing tools.

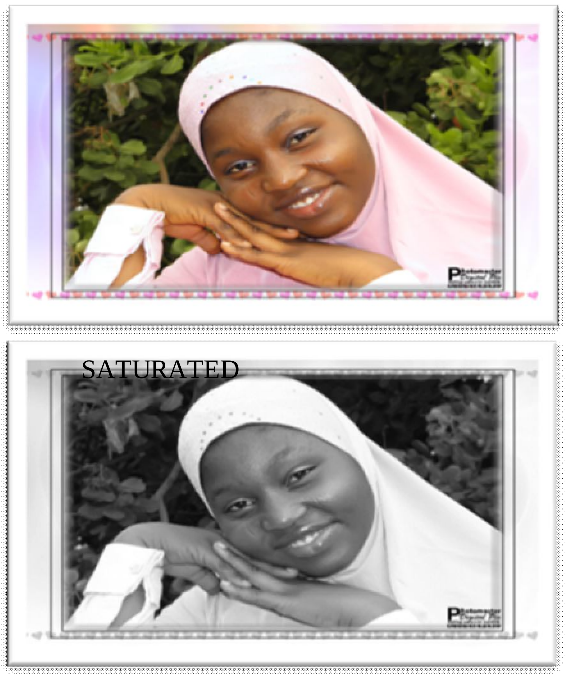
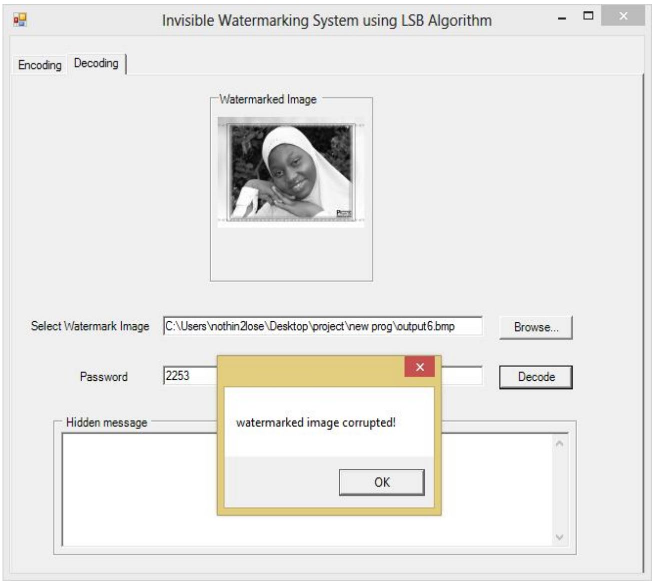


Figure 4.5: Watermarked image with no attack on the left and image with grayscale attack on the right.



Select Watermark Image C:\Users\nothin2lose\Desktop\project\new prog\output6.bmp

Password 2253

watermarked image corrupted!

Figure 4.6: Software displaying error message when the watermarked image attacked with gray scale was to be decoded.

The system displays the "watermarked image corrupted" on trying to decode a grayscale copy of watermarked image that has noise added to it as indicated in figure 4.6.

5.Results and Conclusion

In this paper, a watermarking system was developed using LSB algorithm and the performance evaluation test was performed on the watermarked image after it has been altered graphically (attacked) using distortion, saturation and gray scale noise.

In conclusion, the paper has succeeded in showing the importance of digital media security using the method of watermarking. It has practically shown how digital contents (images in this regard) can be protected from unauthorized use and claims. Tamper detection or unauthorized alterations are also evaluated.

References

[1] Frank H. and Kutter, M. (1999): “Multimedia Watermarking Techniques”, Proceedings on IEEE, 87(7): 1079 – 1107.

[2] Latha, M.M., Pillai, G.K. and Sheela, K.A. (2007): “Watermarking based content Security and Multimedia Indexing in digital Libraries”, International Conference on Semantic Web and digital Libraries (ICSD). ARD Prasad & D. P. Madalli (Eds.).

[3] Cox I, Miller M, Bloom J, Fridrich J, Kalker T (2008): Digital Watermarking and Steganography Second Edition. Elsevier, 2008.

[4] Wang, R. Z., Lin, C. F., Lin, J. C(2003): ‘’Image hiding by optimal LSB substitution and genetic algorithm’’, Pattern Recognition, 34: 671- 683.

[5] Kobayashi M, and Tewfix (1998): “Digital watermarking: Historical roots,” IBM Research, Tokyo Res. Lab., Tech. Rep., Apr. 1998.

[6] Song C, Sud Sudirman, Madjid Merabti (2009): Recent Advances and Classification of Watermarking Techniques in Digital Image