Academic Journal of Science and Technology
ISSN: 2771-3032 | Vol. 5, No. 3, 2023

Research on the Improvement of LSB-based Image Steganography Algorithm

Shuaina Wang[a], Hang Yin, Xiangkun Wang

Computer Science and Software Engineering Department, University of Science and Technology LiaoNing, Anshan, China
[a]E-mail: 2574088230@qq.com

Abstract: This study has made an improvement based on the LSB algorithm, which uses a randomly generated sequence for embedding information, with an implementation environment of Python and a combination of OpenCV By analyzing the embedded data with PSNR values and pretzel-plus-noise attacks, the results show that this method has better image quality and stronger resistance to attacks, while the complexity of extracting the embedded information by external users is high. The improved LSB algorithm in this study has great reference value for information covert transmission, and can be used for image copyright protection and privacy information protection in practical applications.

Keywords: LSB algorithm improvement, Randomly generated sequences, Information embedding, PSNR values, Pretzel-plus-noise attack.

1. Introduction

In today's Internet environment, pictures have become an indispensable part of people's daily life. And yet, a large amount of pictures have been stolen or used unauthorized on the Internet, seriously infringing on the copyright of pictures. At the meanwhile, with the speedy development of artificial intelligence technology, the copyright issue of artificial intelligence-generated images has also gradually surfaced, intensifying the demand for image copyright protection. In order to solve this problem, digital watermarking technology is widely used as an information hiding technology. Digital watermarking technology can protect the copyright of an image while hiding the image, making the hidden information invisible to unauthorised users, which effectively protects the copyright of the image. At the same time, when an image is found to have been misappropriated or used without authorisation, it can be defended by revealing the information embedded in the digital watermark. One of the most commonly used digital watermarking techniques is to embed information by making minor modifications to the image pixels; the LSB (least significant bit) algorithm is a common digital watermark hiding algorithm. However, the robustness of the original LSB algorithm is low and easily detected by attackers. Therefore, this paper proposes an improved LSB algorithm based on random sequences, aiming to improve the security and robustness of the algorithm. Compared with the traditional LSB algorithm, this improved algorithm can protect image copyright more effectively while protecting private information, and is more suitable for information hiding and protection in practical applications. This algorithm can also help to defend the copyright of images when needed.

2. The Principle of the Traditional LSB Algorithm

2.1. Introduction to RGB

RGB refers to the three basic colours of red (Red), green (Green) and blue (Blue) and is one of the most common colour modalities used in computer image manipulation.In RGB mode, the colour of each pixel point is mixed by different proportions of these three basic colours.In an 8-bit RGB image, for example, each pixel can be represented by three 8-bit unsigned integers, representing the red, green and blue constituents, which all take values in the range 0-255, as shown in Figure 1. The different combinations of red, green and blue can present a wide variety of colours, and this makes RGB mode one of the most popular colour modes for computer image manipulation.

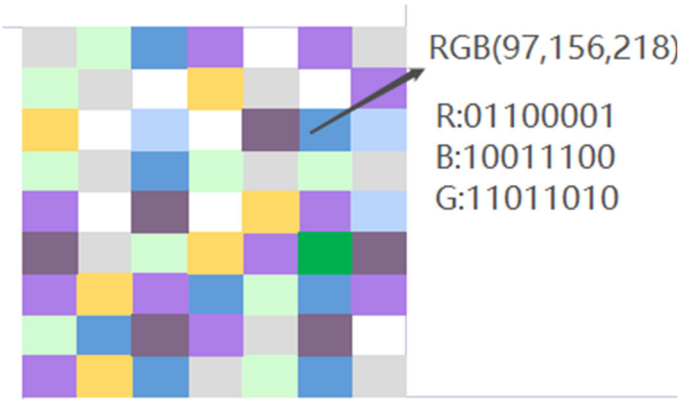

Figure 1. RGB Example

2.2. Introduction to the LSB Principle

The traditional LSB algorithm is a simple method of information hiding. The principle is to embed the information to be hidden in binary form in the least significant bit (LSB) of the image, so as not to cause significant visual differences to the original image, but to protect the privacy and security of the information to a certain extent [1].

In an 8-bit grey-scale image, the value of each pixel can be represented as an 8-bit binary number ranging from 000000 to 111111111, corresponding to decimal numbers from 0 to 255, and it is this least significant bit, the last bit in the binary number, that the LSB algorithm utilises. Since the lowest bit only causes a small change in the pixel value, embedding the information to be hidden in the lowest bit does not have a significant impact on the quality of the image. The image is shown in grey scale (Lena image) and split into 8 bit planes. The image is shown in Figure 2.

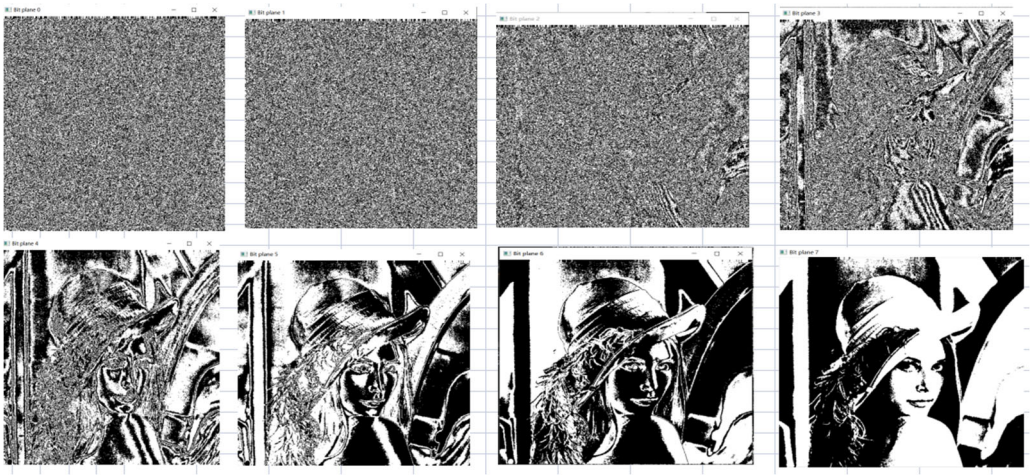

Figure 2. 8 layer bit plane

2.3. Introduction to the process

The emplacement process usually comprised the following procedures.first, the carrier image and the watermarked image are read and they are converted into a three-channel BGR format. Next, the watermarked image is converted into a greyscale image and binarised to obtain a binary matrix. The bit plane of one of the channels is then extracted from the carrier image as the embedding channel and an all-1 matrix of the same size as the embedding channel is constructed. According to the watermark information after binarisation, the position corresponding to the pixel point that needs to be embedded in the watermark is changed to 0. Next, the lowest significant bit of each pixel value in the embedding channel is subjected to the sum operation with the all-1 matrix, and the pixel value with the lowest significant bit position of 0 is changed to 1 or 0 to embed the watermark information. Finally, the embedded channels are reassembled into the carrier image after embedding the watermark, and the final watermarked image is saved locally.

The extraction process commonly comprises the following procedures.first, the watermarked image is read and one of the channels is extracted as the embedding channel. Then, for each pixel value in the embedded channel, the lowest significant bit is extracted to obtain a binary matrix. The embedded watermark information is then extracted from it to complete the extraction process.

3. The Improvement of the LSB Algorithm

3.1. The Embedding Process

The improvement process of the LSB algorithm is very well done by using both Python and OpenCV. Python is widely used in image processing because it is a high-level language with good readability and flexibility. OpenCV is a C++ based computer vision library that also provides a Python API for processing images in Python. First, the hidden and carrier images are read through OpenCV and converted to grayscale, ensuring they have the same dimensions. The hidden image is then converted to a binary stream of numbers and the length of that stream is obtained. Next, The data for the preparation of the embedding is acquired from the number of rows and columns of the carrier image. And a list of all the pixel coordinates of the carrier image is generated for use in the next step. Then, based on the list of all the pixel coordinates of the carrier image, a random list of position sequences is generated using Python's random.sample function. This list contains the pixel positions of the stream of binary numbers to be embedded in the carrier image to hide the image. This list of position sequences is generated randomly to make the embedded data more invisible. When generating the sequence of positions, it should be ensured that each pixel is selected only once to avoid data loss or duplicate embedding. Finally the image to be hidden is embedded into the carrier image according to the coordinates of the previously generated random sequence. Specifically, we need to obtain the bit-plane of each pixel and replace the bit-plane bits with the bit values of the binary digital stream to be embedded at the specified bit-plane. Note that embedding the data should ensure that the lowest valid bits of the pixel are modified to minimise the impact on the carrier image.

3.2. Extraction process

The carrier image with the hidden image embedded in it is retrieved and converted to a grey scale image. The number of rows and columns of the carrier image are obtained and a list containing all the pixel coordinates of the carrier image is generated. Obtain a list of position sequences for the embedded data based on the list of random sequences used previously. This position sequence list contains the pixel positions in which the binary digital stream of the hidden

image is embedded. Extract the binary digital stream of the hidden image from the carrier image according to the position sequence list. The extracted binary digital stream is converted to the previously hidden image. Specifically, the hidden image is obtained by reorganising the binary digital stream into pixel form and converting it to a grey-scale image.

4. Analysis of Results

4.1. PSNR Analysis

PSNR (Peak Signal-to-Noise Ratio) is a metric used to measure the quality of an image and can be used to compare the degree of difference between the original image and the compressed or processed image [2]. PSNR is calculated by the following formula:

$$PSNR = 10\log_{10}\left(\frac{MAX^2}{MSE}\right)$$

MAX there is the maximum number of values possible for an image or video pixel value, usually 255, and MSE is the Mean Squared Error (MSE), which represents the average of the sum of the squares of the errors between the original and the compressed image. MSE is calculated as follows:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[f(i,j) - g(i,j)]^2$$

The better the value of PSNR, the lower the variance between the two images and the better the image quality. Generally speaking, when the PSNR is greater than 30dB, The difference between the two images is hardly recognisable to the human eye.Through testing, it is found that the improved algorithm based on LSB in this paper achieves a PSNR value of 50 or more with high image quality.

4.2. Pepper attack

The pepper attack is a process in which certain pixel points in an image are replaced with white or black noise in digital image processing. These noises look like pretzels, hence the name pretzel noise [3]. This attack is a common form of digital image processing attack that can be used to corrupt the information in an image, degrade the quality of the image and interfere with image-based algorithms. This improved method is resistant to the pretzel attack. It is shown in Figure 3.

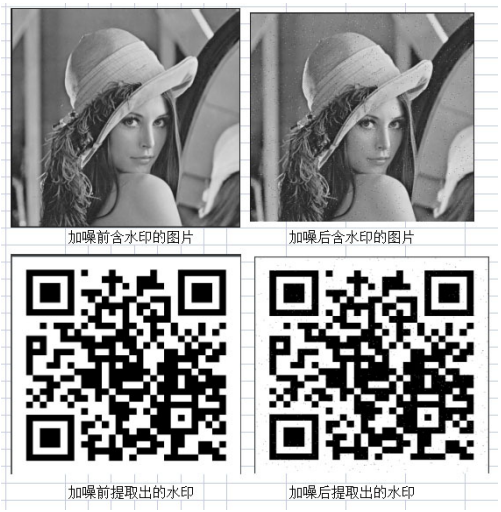

Figure 3. Comparison of extraction effect before and after noise addition

5. Conclusions

In general, the results of this paper show that the improved LSB algorithm based on random sequences can protect image copyright and hide private information more effectively. Compared with the traditional LSB algorithm, the algorithm is improved in terms of security and robustness, and can effectively maintain the quality of images. Therefore, the algorithm has wide application value and can provide strong support for digital copyright protection. However, as artificial intelligence technology continues to develop, digital copyright protection will face increasingly complex challenges. Continuous improvements in digital watermarking technology and more in-depth research are needed to meet the challenges ahead.

References

[1] Sun Mei,Pei Jianhang. Performance analysis of commonly used digital watermarking techniques under attack[J]. Journal of Baicheng Normal College,2023,37(02):46-58.

[2] Shi Xianzhuo,Li Yuandan,He Dan et al. An improved algorithm for steganography based on LSB grayscale image information[J]. Information and Computer(Theory Edition), 2019, 31(21):62-64.

[3] Yin Hao, Liu Bingxing. Digital watermarking copyright protection technology based on two-dimensional Fourier transform [J]. Electronic Technology and Software Engineering, 2021, No.200(06):110-111.