Image Steganography using LSB and DCT algorithm

K Sreekar Reddy(19BCE1227) [#1] P Saiteja (19BCE1211) [#2] Bollineni Nishanth(19BCE1805) [#3]

Abstract

In the past, humans have used a lot of techniques like using invisible ink to conceal or hide information so that only particular people can extract the information present in the paper. During the Revolutionary War invisible ink usually consisted of a mixture of ferrous sulfate and water. The secret writing was placed between the lines of an innocent letter and could be discerned by treating the letter with heat or a chemical substance. As the technology developed hiding techniques also developed in these modern days which use digital images to hide confidential information.

This paper introduces an algorithm of digital watermarking based on Least Significant Bit (LSB), Discrete Cosine Transform (DCT) and also image steganography (hiding image in an image) using Least Significant Bit Substitution technique. According to the characters of human vision, in this algorithm, the information of digital watermarking which has been discrete Cosine transformed, is put into the high frequency band of the image which has been wavelet transformed. Then distills the digital watermarking with the help of the original image and the watermarking image. The simulation results show that this algorithm is invisible and has good robustness for some common image processing operations.

1. Introduction

With the redundancy of the medium as image and voice, digital watermarking technology is to use the digital embedding method to hide the watermarking information into the digital products of image, visible and video. Seen form the field of signal process, the watermarking signal being embedded into carrier is as a feeble signal to add into a strong background. As long as the intensity of watermarking is lower than the contrast restriction of human visible system (HVS) or the apperceive restriction of human audio system (HAS), the watermarking signal won't be felt by HVS or HAS. With the characters and important application, digital watermarking technology has been got more and more attention. In the future the main development of digital watermarking is like this: copyright protection, pirate tracking, copying protection, image authentication, cover-up communication, classification control of digital watermarking video and so on. And the common characters of digital watermarking are: insensitivity, secrecy, robustness and insurance. According to the different partitions, watermark can be parted in different types like these: significant watermark and the insignificant; the visible and the invisible; the brittle and the steady; the spatial domain watermark and the transformed domain watermark; the blind, the semi blind and the nonblind. One another partition is carrier and there is image watermark, audio watermark, video watermark, text watermark and so on. The current classical algorithm contains spatial domain algorithm and transformed domain algorithm. With the spatial domain algorithm, the embedding and the distilling of watermarking are finished in spatial domain, by amending directly or comparing the gray-level value or color value. The classical spatial domain algorithms including several ways as follow: the least significant bit (LSB), Patchwork method with streak block mapped coding, the method based on district intersecting and so on. Then the main current transformed domain algorithms are spread spectrum, DCT transformation method. This paper introduces an algorithm of digital watermarking based on Least Significant Bit (LSB), Discrete Cosine Transform (DCT). The watermarking image will be discrete Cosine transformed at first. Because these DCT modulus contain the low frequency information of watermarking image, as long as this information do not lose or lose little then the watermarking image can be renewed well. This enhances the robustness and concealment. We have used Python 3.8 for making the project. We used PIL for Spatial domain transformations like LSB and OpenCV for Frequency domain transformations like DCT implementations. We are hiding text inside a carrier image and make that stego to be decoded later and get the hidden text back. We stored MSE and PSNR for LSB, DCT and stored them in an excel spreadsheet.

2. Literature Survey

Article1:

Link:
https://www.researchgate.net/publication/335228119_An_Image_Steganography_Algorithm_using_LSB_Replacement_through_XOR_Substitution

"Least Significant Bit replacement algorithm, is the most popular and widely used technique in image steganography due to its simplicity and effectiveness. Different methods of data hiding in spatial domain have been proposed and continue to be improved upon. Among them, the LSB replacement method is quite simple and the most popular. However, because the LSB replacement method is quite simple, compared to the other methods, some of its security issues must be improved upon. This paper proposes a highly secured data hiding technique in the spatial domain of image steganography. The proposed scheme takes the message bit and performs XOR operation with the 7th bit of every RGB component and, after then, the produced output is embedded within the 8th bit of each component of RGB. The embedding procedure is done in a way that there will be no sign of original message inside the cover object and, obviously, without using any outside key."

Article2:
Link:
https://www.researchgate.net/publication/330565811_Hiding_data_in_images_using_DCT_steganography_techniques_with_compression_algorithms

"In this article, technique applied is that multiple methods to hiding image by applying DCT algorithm to embed the image and encrypt the data via cryptography algorithms, applying DCT compression, then stego-image will transfer via the internet after that on the other side the reverse method will be performed on stego-image by decrypting it using the private key to extract the data. In the proposed algorithm, it is assumed that the sender as well as the receiver holds the same system of private keys. Indeed, the receiver sends the public key to the sender by an insecure communication channel. Then, the sender generates the stego-image with both keys and sends them through another insecure channel to the receiver, who can extract the secret file, which inserted into the cover image by the embedding procedure."

Article -3:
Link:
https://www.researchgate.net/publication/224074162_Analysis_of_LSB_based_image_steganography_techniques
"In this paper, we have taken the binary representation of the hidden information and overwrite the LSB of each byte within the cover image. Here we have introduced a secret key to protect the hidden information.
The following formula, we have used in our proposed method is:
cover image + secret key + hidden information = stego image"

3. Proposed Work - Algorithm; Justification;

LEAST SIGNIFICANT BIT TRANSFORM

The least significant bit (in other words, the 8th bit) of some or all of the bytes inside an image is changed to a bit of the secret message. Digital images are mainly of three types:

i. 8-bit images
ii. 24-bit images
iii. 32-bit images

In 24-bit images we can embed three bits of information in each pixel, one in each LSB position of the three eight-bit values. Increasing or decreasing the value by changing the LSB does not change the appearance of the image; much so the resultant stego image looks almost same as the cover image. In 8-bit images, one bit of information can be hidden.

The hidden image is extracted from the stego-image by applying the reverse process. If the LSB of the pixel value of cover image C(i,j) is equal to the message bit m of secret massage to be embedded, C(i,j) remain unchanged; if not, set the LSB of C(i, j) to m. The message embedding procedure is given below

S(i,j) = C(i,j) - 1, if LSB(C(i,j)) = 1 and m = 0
S(i,j) = C(i,j), if LSB(C(i,j)) = m
S(i,j) = C(i,j) + 1, if LSB(C(i,j)) = 0 and m = 1

Where LSB(C(i, j)) stands for the LSB of cover image C(i,j) and m is the next message bit to be embedded. S(i,j) is the stego image. As we already know each pixel is made up of three bytes consisting of either a 1 or a 0.
For example, suppose one can hide a message in three pixels of an image (24-bit colors). Suppose the original 3 pixels are:
(11101010 11101000 11001011)
(01100110 11001010 11101000)
(11001001 00100101 11101001)

A steganographic program could hide the letter "J" which has a position 74 into ASCII character set and have a binary representation "01001010", by altering the channel bits of pixels.

(11101010 11101001 11001010)
(01100110 11001011 11101000)
(11001001 00100100 11101001)

In this case, only four bits needed to be changed to insert the character successfully. The resulting changes that are made to the least significant bits are too small to be recognized by the human eye, so the message is effectively hidden. The advantage of LSB embedding is its simplicity and many techniques use these methods. LSB embedding also allows high perceptual transparency

DISCRETE COSINE TRANSFORM

DCT coefficients are used for JPEG compression. It separates the image into different parts of importance. It transforms a signal or image from the spatial domain to the frequency domain. It separates the image into high, middle and low frequency components. In low frequency sub-band, much of the signal energy lies at low frequency which contains most important visual parts of the image, while in high frequency sub-band, high frequency components of the image are usually removed through compression and noise attacks. So, the secret message is embedded by modifying the coefficients of the middle frequency sub-band, so that the visibility of the image is not affected.

With the character of discrete Fourier transform (DFT), discrete cosine transform (DCT) turns over the image edge to make the image transformed into the form of even function. It's one of the most common linear transformations in digital signal process technology. Two-dimensional discrete cosine transform (2D-DCT) is defined as

$$F(jk) = a(j)a(k)\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f(mn)\cos\left[\frac{(2m+1)j\pi}{2N}\right]\cos\left[\frac{(2n+1)k\pi}{2N}\right]$$

The corresponding inverse transformation (Whether 2DIDCT) is defined as

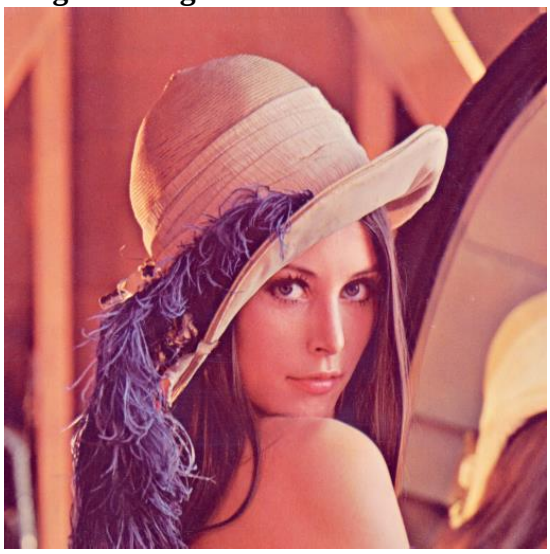
$$f(mn) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} a(j)a(k)F(jk)\cos\left[\frac{(2m+1)j\pi}{2N}\right]\cos\left[\frac{(2n+1)k\pi}{2N}\right]$$

The 2D-DCT can not only concentrate the main information of original image into the smallest low frequency coefficient, but also it can cause the image blocking effect being the smallest, which can realize the good compromise between the information centralizing and the computing complication. So, it obtains the wide-spreading application in the compression coding.

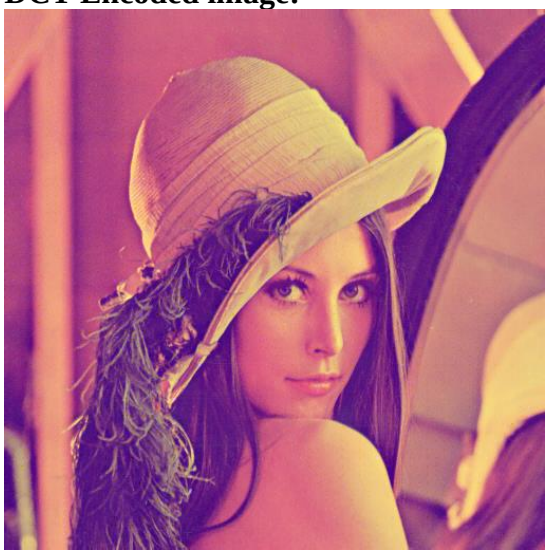
4. Experimental Setup

4.1 DATASET

Original image:



DCT Encoded image:



LSB Encoded Image:

4.2 Software Needed:

The algorithm has been implemented in python which uses the following libraries:

1. **cv2**: OpenCV-Python is a library of Python bindings designed to solve computer vision problems.

2. **PIL**: The Python Imaging Library adds image processing capabilities to your Python interpreter. This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities. The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool.

3. **Matplotlib**: Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

4. **NumPy:** NumPy can be used to perform a wide variety of mathematical operations on arrays. It adds powerful data structures to Python that guarantee efficient calculations with arrays and matrices and it supplies an enormous library of high-level mathematical functions that operate on these arrays and matrices.

4.3 Result and analysis:

ERROR CALCULATION

The following two error metrics are used in the performance analysis.

A. Mean Square Error (MSE): It is defined as the square of error between cover image and the stego image.

The distortion in the image can be measured using MSE

$$MSE = \frac{1}{MN}\sum_{j=1}^{M}\sum_{k=1}^{N}\left(x_{j,k} - x'_{j,k}\right)^2$$

M and N are the number of rows and column in the input image.

B. Peak Signal to Noise Ratio (PSNR):

It is the ratio of the maximum signal to noise in the stego image.

$$PSNR = 10\,log\frac{(2^n - 1)^2}{MSE}$$

$$PSNR = 10\,log\frac{(255)^2}{MSE}$$

PSNR is measured in decibels (dB). PSNR is a good measure for comparing restoration results for the same image.

Performance analysis:

Method	LSB	DCT
Invisibility	Medium	High
Payload Capacity	High	Medium
Robustness	Low	Medium
PSNR	High	Medium
MSE	Low	Medium

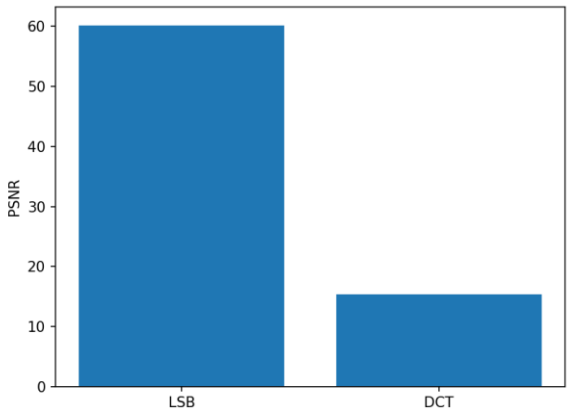
The watermark is embedded in the DCT domain of an image in a multi-resolution way. The DCT scheme also employs visual masking to guarantee that the embedded watermark is invisible and to maximize the robustness of the hidden data. In the decoding phase, once the watermark is extracted from the watermarked image, certain performance measures such as peak signal to mean noise ratio (PSNR) and correlation are calculated. Different types of attacks have been applied to the watermarked image to test the robustness of the applied technique and for each case, PSNR and correlation are calculated.

Our project implementation shows the result for Lenna image:

Original image vs	MSE	PSNR
LSB	0.062302	60.18581
DCT	1888.258	15.37019

Above table shows PSNR for original cover image with after LSB and DCT transformation.

Mean Square Error for DCT is quite higher than LSB, but payload capacity, and robustness is greater in DCT than of LSB.



The previous chart shows the PSNR for LSB and DCT. Peak Signal to Noise Ratio for LSB is quite higher than DCT. Greater the PSNR, lesser the noise in the image. Like if we take 2 identical image and calculate MSE and PSNR, MSE becomes 0, and as PSNR is inversely proportional to MSE, the PSNR becomes infinite.

5. Conclusion

This paper discusses in detail about the LSB, DCT algorithms on steganography application. The LSB, DCT algorithms are implemented for steganography application. In this experiment, performance analysis of LSB and DCT methods are successfully completed and experimental results are discussed. The MSE and PSNR values are compared for the LSB and DCT algorithms. The PSNR value shows the quality of image after embedding the data. From the experiment results it is observed that the PSNR of DCT is high as compared to the other algorithm. Thus, the experiment concludes the DCT algorithm is more suitable for the steganography application compared to the LSB algorithm

6. References

[1] M. H. Abood, “An efficient image cryptography using hash-LSB steganography with RC4 and pixel shuffling encryption algorithms,” in 2017 Annual Conference on New Trends in Information and Communications Technology Applications, NTICT 2017. IEEE, mar 2017, pp. 86–90

[2] K. Joshi, R. Yadav, and S. Allwadhi, “PSNR and MSE based investigation of LSB,” in 2016 International Conference on Computational Techniques in Information and Communication Technologies, ICCTICT 2016 - Proceedings. IEEE, mar 2016, pp. 280–285.

[3] “An Image Steganography Algorithm using LSB Replacement through XOR Substitution” Conference Paper · August 2019
DOI: 10.1109/ICOIACT46704.2019.8938486