Computer Science and Engineering 2021, 11(3): 41-48
DOI: 10.5923/j.computer.20211103.01

Enhanced Error Correcting Code on Information Extraction of Message from Digital Image Using Parity Error Correction

Tebepah T.[1,*], Onuodu F. E.[2], Okolai D. B.[3]

[1]Department of Computer Science, Ignatius Ajuru University, Port Harcourt, Nigeria
[2]Department of Computer Science, University of Port Harcourt, Port Harcourt, Postcode, Nigeria
[3]Department Computer Science, Niger Delta University, Nigeria

Abstract Sending hidden messages through images has a problem of delivering error messages to the recipients. One of the ways that has been used to correct these error messages is by a process called iterating embedding and extraction. In this work, an enhanced error correction code on information extraction from images using parity test has been developed and implemented. The proposed system was built on the existing iterating embedding error correction by adding an additional error check called the Parity correction code. The Object Oriented System development methodology (OOSDM) approach was used for in this project. The system was implemented using C++ and hypertext preprocessor (PHP) for the interactive interfaces. The proposed system had an error free message delivered to the recipient at a more optimal time when compared to the system that used the iterating embedding and extraction process. This system will be of relevance to Enterprises and individuals that use cloud computing for storing information.

Keywords Parity test, Iterating embedding, Extraction

1. Introduction

Storing information on the cloud is a technology that has been around for decades. Large companies and individuals have got over their skepticism of storing information on the cloud. The ease of storing information on the cloud or the absence of a better alternative to cloud computing has influenced the decisions of Companies and Individuals. When companies release their information to the cloud, it is beyond them to safely guide their information [15]. The insecurity that arises adding an error check, called: the Parity correction code. The Object-Oriented System development methodology (OOSDM) approach was adopted in this project. The system utilized the C++ and hypertext preprocessor (PHP) for the interactive interfaces. The proposed system had an error-free message delivered to the recipient at a more optimal time when compared to the system that used the iterating embedding and extraction process. This system will be of relevance to Enterprises and individuals that use cloud computing for storing information. from saving information on the cloud has become a massive problem, especially in this era of cyber attacks. To cub cyber attacks, sensitive information is hidden in an image and sent over the cloud. More so, additional problems come with the extraction of the hidden message from the image. This literature proposes a system that enhances the correction of secret messages extracted from a host image using the parity correction code. There is already existing literature that dealt with a distorted extracted message using the iterating embedding procedure [2]. This article builds on this procedure by adding an error-correcting code (ECC) called parity ECC to the already existing iterating embedding and extraction procedure.

2. Literature Review

Evsutin et al., [2] in their paper: "Algorithm of error-free information embedding into the DCT domain of digital images base on the QIM method using adaptive masking of distortion", tried to address the problem that came with error occupancies that appeared in secret messages extracted from a digital image using a cosine based algorithm. In addition, this paper showed that errors were corrected with the iterative embedding procedure. While their algorithm attempted to extract error-free secret messages from a digital image, it could not do it efficiently.

Arunkumar et al. [15] proposed a robust steganographic method to secure medical images. Although the method was robust and able to give a better level of imperceptibility using

[*] Corresponding author:
tariuget.tt@gmail.com (Tebepah T.)

Received: Oct. 19, 2021; Accepted: Oct. 30, 2021; Published: Dec. 24, 2021
Published online at http://journal.sapub.org/computer

SVD and DCT, They could not cover the embedding of secret messages on the medical image block based on the contrast and correlation measure.

Evsutin & Kokurina [14], in their paper "embedding into digital images based on the discrete Fourier transformation", proposed a new algorithm for embedding messages into digital images using discrete Fourier transformation. Although their work addressed the distortion problem with extracted messages and gave a high-quality steno-image, they could not improve the embedding quality by including other optimisation methods.

A JPEG picture that has undergone compression twice will be nonaligned, and estimating the first quantisation matrix is a problem. Yao et al. [5] attempted to address this problem in their article: "improved first quantisation matrix estimation for nonaligned double compressed JPEG images". Although they were able to achieve this in two phases: by estimating the parameters that were misaligned and estimating the quantisation step, they, however, used numerous empirical, theoretical thresholds which were of a distinct theoretical basis.

Forgers can re-encode high definition (HD) videos to a lower quality video by decreasing the quantisation parameter (QP) or increasing the bit rate of the video. "Detection of fake high definition for HEVC videos based on prediction mode feature" addressed this problem [16]. The authors proposed a method to detect fake videos over highly efficient video coding (HEVC) by Prediction Mode Feature (PMF). They claimed that their method was better than the already existing method. On the other hand, their work did not cover detecting fake videos on social networks or video websites.

Li et al., [6], in their paper: "Extracting Spread-Spectrum Hidden Data from Digital Media", Proposed a multicarrier iterative generalised least square algorithm (M-IGLS) that gave an error probability that was similar to techniques used to solve the extraction problem. On the other hand, their algorithm was not able to give an error-free extraction process. It was more like reinventing the wheel.

Yao et al., [1], in their article "High-fidelity dual-image reversible data hiding via prediction-error shift", enhanced an already existing dual-image reversible data hiding (RDH) technology. Their experiment came out with a method they claimed was highly reliable and efficient. Although they claimed that their experiment was successful, it did not meet the requirements of a high-fidelity situation as it should.

Chen et al. [3] addressed the tradeoff problem that arose from embedding a signal within a host to form a compound signal with their classes of new methods. Although their method was sustainable against attacks on copyright applications, their Quantisation Index Method (QIM) was termed "close to optimal" by the authors.

Porat [7], in their article "Streaming k-mismatch with error correcting and application", presented an efficient design streaming algorithm for series of problems that required an approximate pattern matching Although they were able to achieve an efficient design for pattern matching, more efficient research on K-match with error-correcting was being made concurrently.

Lugosch [8] Their thesis titled "Learning Algorithms for Error Correction" attempted to address the complexity that came with short block codes by proposing a recast of the short block code using machine learning language. They claimed that the modification proposed by the thesis improved the neural belief propagation algorithm in performance and reduced its implementation complexity.

Dziech & Wassermann [9], in their paper "Application of Enhanced Hadamard Error Correcting Code in Video-Watermaking and his comparison to Reed-Solomon Code", introduced an enhanced variant of the Hadamard Error Correcting Code (EHC). The Enhanced Hadamard Error-correcting Code (EHC) was applied to video watermarking. Watermarking could not survive high video compressing. However, with the use of EHC, the watermarking was salvage after the compression process.

Wootters [10], In her seminar titled "Plus Group Testing", established that repetition of the message is expensive; it cost time, power. She proffered other ways of error correction instead of message repetition. Furthermore, the seminar talk defined error-correcting codes, what they are, and how they can solve problems in communication. Although the speaker spoke extensively on error-correcting codes, she laid more emphasis on Reed Solomon's code.

Ashura [11], In his article "Error Minimization in BCH Codes," proposed a coding scheme he claimed reduces complexity and increases efficient error performance compared to the frequently used scheme. The researcher used BCH to encode and decode a text message. Their suggestion for future research was to apply optimisation to the speed and device by parallel methodology.

Sun et al., [12], In their review "A secure and robust approach to scalable video authentication," presented a content authentication scheme with three video transcoding methods that the authors felt can manipulate the contents of a video. The methods are used when the bit rate of the streaming is to be reduced. The methods are frame resizing, frame dropping and multi-cycle coding. They said the error-correcting codes (ECC) could handle the random errors. More so, despite the distortions and transcoding infrastructure, they claimed that their scheme ensured end-to-authentication.

Jiang et al., [13], In their paper "Enhanced Error Correction via Language Processing", talked about two basic approaches for error correction. One of the approaches was the use of redundancy inside data after compression. The other approach was to add external redundancy to the data. Their article focused on the second approach and how it can enhance error correction performance. They argued that using ECC and internal redundancy are compatible because data losing redundancy due to ECC does not affect the data integrity.

Jin et al., [4], in their paper "Feature extraction optimisation of JPEG steganalysis based on residual Images", proposed a novel scale adaptive algorithm for the feature extraction filter according to the JPEG factor for optimal performance. Although they got their required result, their proposed algorithm dealt with JPEG images because other picture formats exist.

3. Analysis and Materials

This section aims to analyze the existing system and the concept design of this article. This analysis will serve as a yardstick for evaluating the proposed system.

3.1. Discrete Cosine Transform (DCT) Procedure

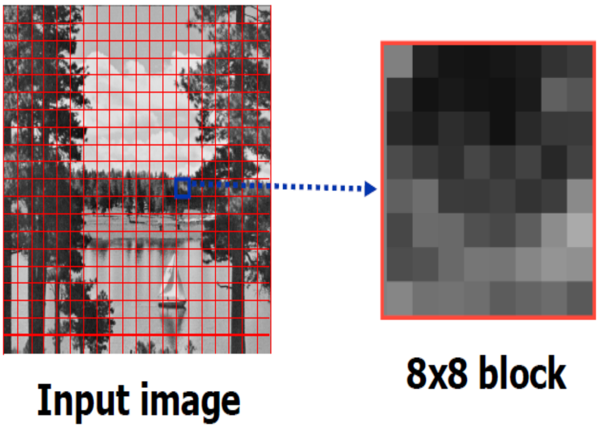


Figure 3.1. Input image (Source: Khana, 2009)

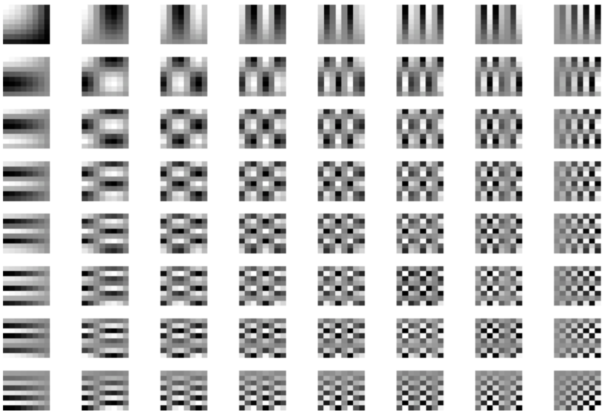


Figure 3.2. Cosine wave (Source: http://en.wikipedia.org/wiki/JPEG)

$$\overset{\overset{x}{\longrightarrow}}{\begin{bmatrix} -76 & -73 & -67 & -62 & -58 & -67 & -64 & -55 \\ -65 & -69 & -73 & -38 & -19 & -43 & -59 & -56 \\ -66 & -69 & -60 & -15 & 16 & -24 & -62 & -55 \\ -65 & -70 & -57 & -6 & 26 & -22 & -58 & -59 \\ -61 & -67 & -60 & -24 & -2 & -40 & -60 & -58 \\ -49 & -63 & -68 & -58 & -51 & -60 & -70 & -53 \\ -43 & -57 & -64 & -69 & -73 & -67 & -63 & -45 \\ -41 & -49 & -59 & -60 & -63 & -52 & -50 & -34 \end{bmatrix}} \Big\downarrow y$$

Figure 3.3. Image Shifted by -128 (Source: http://en.wikipedia.org/wiki/JPEG)

$$\overset{\overset{u}{\longrightarrow}}{\begin{bmatrix} -415 & -30 & -61 & 27 & 56 & -20 & -2 & 0 \\ 4 & -22 & -61 & 10 & 13 & -7 & -9 & 5 \\ -47 & 7 & 77 & -25 & -29 & 10 & 5 & -6 \\ -49 & 12 & 34 & -15 & -10 & 6 & 2 & 2 \\ 12 & -7 & -13 & -4 & -2 & 2 & -3 & 3 \\ -8 & 3 & 2 & -6 & -2 & 1 & 4 & 2 \\ -1 & 0 & 0 & -2 & -1 & -3 & 4 & -1 \\ 0 & 0 & -1 & -4 & -1 & 0 & 1 & 2 \end{bmatrix}} \Big\downarrow v$$

Figure 3.4. DCT block (Source: http://en.wikipedia.org/wiki/JPEG)

The Images to be embedded are partitioned into eight by eight distinct blocks of the exact sizes, as shown in figure 3.1. Each of the eight by eight blocks is encoded with its discrete cosine transform. Before applying the DCT on the block, the values in each block are shifted by -128 to surround zero. The DCT procedure calculates the wave coefficient.

3.2. The Quantisation Stage

Removing the high-frequency data is called quantisation. There is a standard Jpeg quantisation table that represents 50 per cent picture quality. The process involves dividing each coefficient from the DCT table by the corresponding value on the quantisation table and round the value to the nearest integer.

3.3. The Embedding Stage

The message to be sent through the picture is disintegrated into workable lengths, and they are in bits. Likewise, the image is broken down into precise sizes. These broken image sizes are subjected to a frequency transform; at this point, it is ready to receive the hidden message. Note that the number of message fragments to be sent is equal to the number of broken images or blocks. The blocks are processed one after the other by adding the message into the block. An inverse frequency is applied to the processed block, and the resultant image is a stego-image pixel.

The embedded phase is summarised as follows:

Step 1: the picture image is decomposed into non-overlapping blocks of 8x8 pixels.

Step 2: it is assumed that the message embedded in the picture is already in its bit state for simplicity.

Step 3: the message is embedded into the first non-empty box of the image.

The area on the block being considered for embedding is converted into bits and added to the message. Given that n = embedded area and m = message. The embedding starts at the non zero blocks of the picture. This equation that shows how m is added to n is shown below:

$$M=m_1m_2\ldots m_{n-1}n_1m_{n+1}\ldots\ldots m_{2n}\text{-}_{2}n_2\ldots\ldots m_{4n-3}m_{4n-2}\ldots\ldots\ldots m_l$$

Input: Picture image (JPEG) with a size of 512 x 512
Output: Stego Image with a size 512 x 512

4. Current System Assessment

The Stego-image from the embedding phase passes through a data extraction test and an error checking test. There are cases where embedding does not take place. Situations like this will require that the embedding process repeatedly occurs. In addition, the extracted message may come with errors at this point. A situation like this will warrant the re-embedding process again. The message block passes through a frequency transform, and a new stego-image is produced again. The embedding process repeatedly occurs until an error-free message is obtained or not. The re-embedding process done to obtain error-free messages is called the Iterative embedding procedure. Another case might be that of a message that continuously gives an error-free extraction. This message block in this category will be termed an empty message block, and its final interpretation will be a message block that had no bits embedded in it.

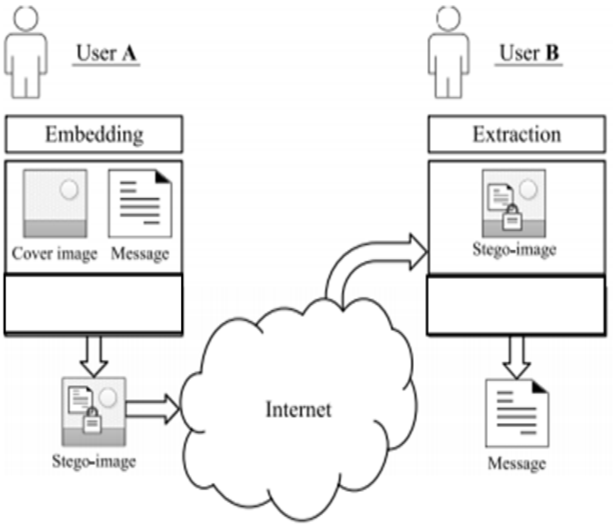


Figure 4.1. Existing model of the extraction process of a message from a Stego-image (Source: Evsutin et al., 2021)

4.1. Existing Algorithm

```
Input: non-compressed grayscale image I; embedding area ρ; coefficient-flag coordinates in DCT block (u, v).
Divide image 'I' into not-overlapping blocks Ir of the size 8 × 8.
k ← 1, l ← 1.
while l < 6 do
    Ck = DCT(I k ).
    Calculate the quantisation step qk and the histogram of quantisation step values HK
    if k > 1 and hk (qk − 1) < hk (qk) then
        Calculate Qk anew and HK
    end if
    f ← Extract a bit from element cuv with quantisation step qk
    if f = 1 do
        nl ← Extract a bit from the last element of the embedding area of block Ck with quantisation step qk.
        l ← l + 1.
    end if
    kk + 1.
end while
n ← Transform binary sequence n1n2n3n4n5n6 into a decimal number.
s ← 1.
for each block, (I r) do
    Cr = DCT(I r).
    Calculate the quantisation step QR and the histogram of quantisation step values hr
    if r > 1 and hr(qr − 1) < hr(qr) then
        Calculate QR anew
    end if
    f ← Extract a bit from element cuv with quantisation step qr.
    if f = 1 do
        Ms ← Extract a message fragment with the length of n bit from the embedding area of block Cr with quantisation step qr.
        s ← s + 1.
    end if
end for
M ← Form message M from extracted fragments Ms .
Output: message M.
```

4.2. Advantages of Existing System

1) Extraction of messages from images is a comprehensive and detailed process.
2) Images devoid of messages are identified and adequately processed so as not mistakenly interpreted as images that need to be re-embedded and re-extracted.

4.3. Advantages of Existing System

The disadvantage of the existing system are as follows:

1) Excessive time delay in getting an error-free extracted message from the image through several iterating processes.
2) There are instances when blocks of images delivered for further processing are without any embedded message.

5. Proposed System

An extra bit of '0' or '1' is attached to the binary message before embedding in parity error correction. We have two kinds of parity: even parity and odd parity. If the recipient is expecting a message encoded with even parity but gets an odd number of '1s', it is an error indicator. However, if the recipient expects an error encoded in odd parity but received an even number of '1s', this is another error indicator.

Assuming we have a '10011010' message to be sent through an image. The number of '1s' in this code is 4. If the message is modified using even parity, a '0' is added at the end of the coded message making it 100110100. Conversely, when the message is delivered using the odd parity, a '1' is added at the end of the code, increasing the number of '1s' to 5.

The algorithm aims to embed the message in a bit into the eight by 8 DCT block into selected sub-blocks represented by coordinates x and y. Even parity is used on the message bit so that the recipient will expect a message even after extraction. The iterative process and the parity error-correcting code does the error checking. The iterative process (τ) repeats itself a maximum of three times to get three different outputs which the parity error-correcting code would correct. The‘s’ variable is a counter for the message being embedded, ‘p’ is a counter for extracted error messages, it is the counter for the iteration. Its default setting is 0 whenever the algorithm starts up with another block. The D' and D'' is used to check the transition states of the process DCT block. Another essential variable is the 'α' variable which monitors the present result of the present DCT block processing. The α variable can exist in three different states: 0, 1, and 2.

1) When α = 0, it means the DCT block processing has commenced.
2) When α = 1, it means there are errors on the embedded message fragment. One such error is the inversion of one or more of the bits.
3) When α = 2, it means an error-free image has been extracted.

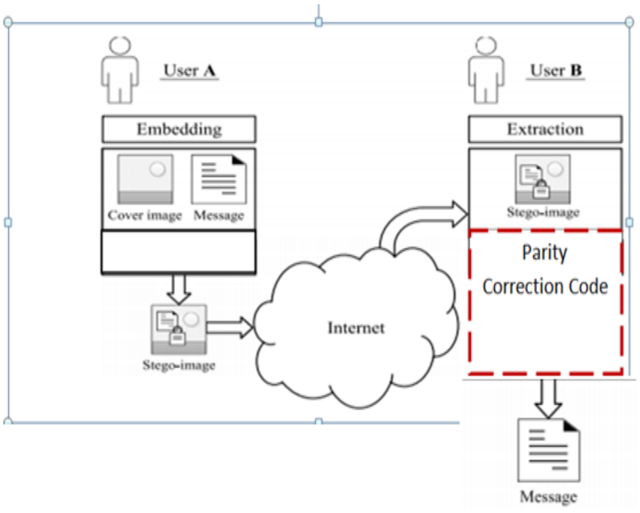


Figure 5.1. Proposed Model for Error Correction for Extracted Message using Parity Correction Code

5.1. Proposed Algorithm

Input: non-compressed grayscale image I; embedding area ρ; coefficient-flag coordinates in DCT block (u, v).

```
Divide image I’ into not-overlapping blocks Ir of the size 8 × 8.
k ← 1, l ← 1.
while l < 6 do
    Ck = DCT(I   k ).
    Calculate the quantisation step qk and the histogram of quantisation step values HK
    if k > 1 and hk (qk − 1) < hk (qk ) then
        Calculate qk anew   and hk
    end if
    f ← Extract a bit from element cuv with quantisation step qk.
    if f = 1 do
        nl ← Extract a bit from the last element of the embedding area of block Ck with quantisation step qk.
        l ← l + 1.
    end if
    kk + 1.
end while
n ← Transform binary sequence n1n2n3n4n5n6 into a decimal number.
s ← 1.
for each block, (I r) do
    Cr = DCT(I   r).
    Calculate the quantization step qr and the histogram of quantization step values hr
    if r > 1 and hr(qr − 1) < hr(qr) then
        Calculate qr anew
    end if
    f ← Extract a bit from element cuv with quantization step qr.
    If   (f = 1)
        {
            Extract a message fragment with the length bit from the embedding
            area of block Cr according to formula (2) with step q_r.
            s++ //increment s counter
        }
    // the code below executes the extraction 3 times with 3 error message output
    For (b = 0; b >= 2; b++)
    {
        If (f = 2)
        {
            Extract error message
            p++ // increment u counter, u counter keeps track of the error message loop
        }
    }
    Use Ecc code to correct the 3 message output
    M← Form message M from extracted fragments Ms
    Output: message M
```

5.2. Methodology

The Object-Oriented System Development Methodology (OOSDM) was adopted to implement the steganography extraction procedure. OOSDM is robust and was used to produce the analysis model and the design model. There were four major classes: the Image class, Message class. Embed class and the Extract class. Their objects were made to interact with each other through their methods and functions. The image and the message instances were implemented in the embed class. At the point of extraction, the Image and Message instances were implemented in the extraction class. The parity correction code was implemented as a function in the Message class, where the message could pick between an even parity or an odd parity. In addition, the scope of the parity function was extended to the extraction process because it was required to check if the extracted messages complied with the chosen parity type. The rationale behind choosing OOSDM was because it supported the interactions of objects to form another composite object. Most applications presently use OOSDM. For this article, the OOSDM was applied in-network and cybersecurity. The stego-image has been through the internet to the recipient. The recipient had expected a code that can extract the message from the image. It is a bit similar to sending a zip file through the internet and the recipient unzipping the zip file.

5.3. Advantages of the Proposed System

1) There is a greater likelihood to get a more accurate message after extraction.
2) The iteration embedding process is limited to three iterations because an additional error checking system has been included in the overall process.
3) The time taken for the extraction process is reduced since the number of embedding iteration is being controlled.

6. Output and Discussion

Figure 6.1 shows three authorised users having access to the secure stego images while the fourth person is denied access. Each authorised user can send messages in the images to the sender, and they can also retrieve the senders' hidden message through the download and extract button in figure 6.3. The proposed interface will have an additional button called Parity Check. When the message has been retrieved through the download and extract button, an additional error check will be made by clicking the Parity check button. All authorised users will have login details which will be entered into a GUI like Figure 6.2. The interface of all authorised users will look like the image in figure 6.3 after log in. unauthorised users will not be able to access this interface. Each user will have the option of downloading the image from the cloud using the download button, as seen in Figure 6.3. Furthermore, once the image is downloaded from the cloud, the message embedded in it will be extracted, and the hidden message is separated from the image, as seen in figure 6.5. The hidden messages will usually come with errors. The message will be subjected to a parity test where the error in the message is identified and corrected (Figure 6.5-6.8).

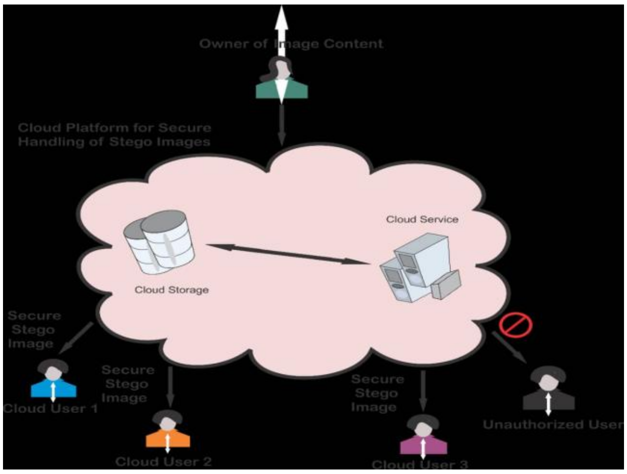


Figure 6.1. Authorised users and Unauthorised users of the cloud services

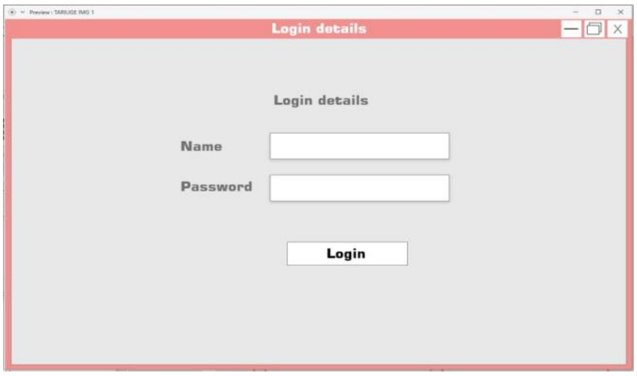


Figure 6.2. User Interface for authorised user

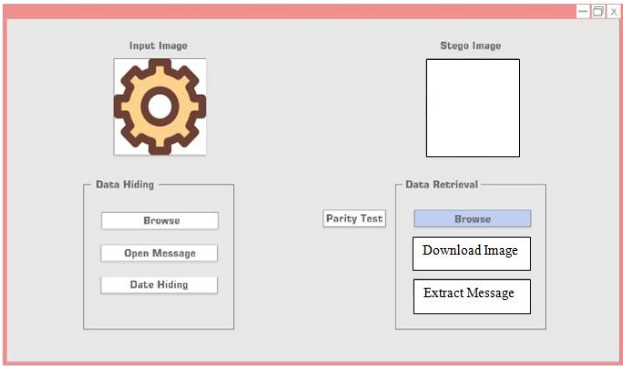


Figure 6.3. Home Page Interface after login

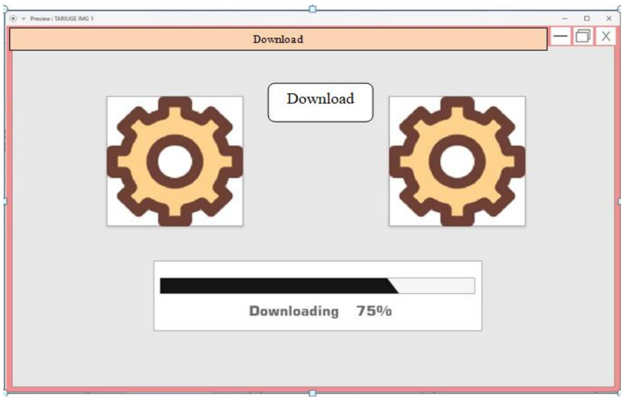
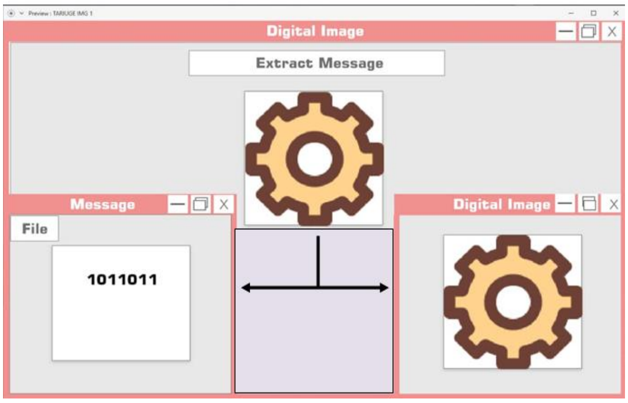


Figure 6.4. Downloading Image from the cloud



1011011

Figure 6.5. Extract message from the image

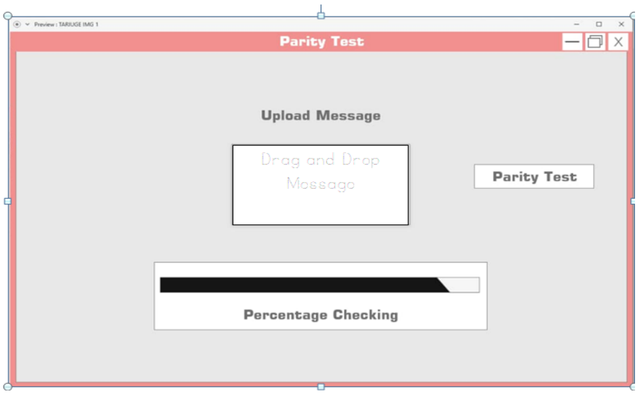


Figure 6.6. Parity Test

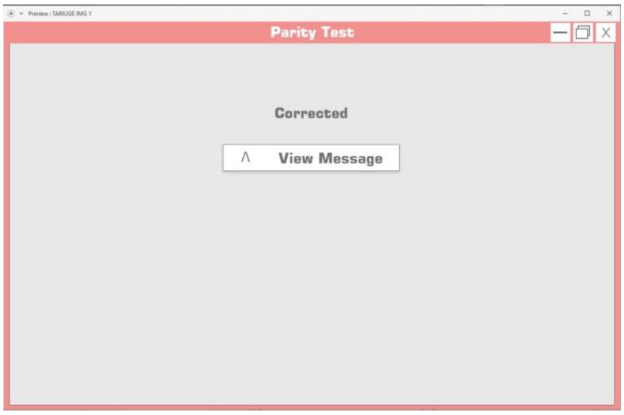
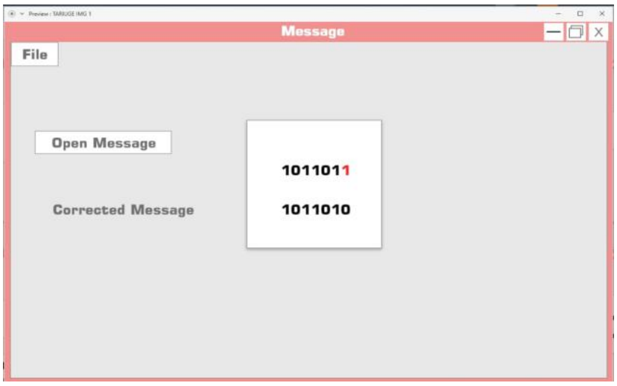


Figure 6.7. Parity Test completed, the corrected message will be accessed through the 'View Message' button



1011011

1011010

Figure 6.8. Corrected message

7. Conclusions

In this paper, we looked into the entire process of hiding a message in an image using a discrete cosine transform. The embedding procedure was implemented by using the Quantisation index modulation (QIM). The iterating embedding procedure was used to extract the message from the image to achieve an error-free message. The author added an error-correcting measure to the iterating embedding procedure in the form of parity correcting codes. There are tones of error-correcting codes in the literature. Researchers have shown that some of the error-correcting codes are more efficient than the Parity ECC. Futuristically researchers may want to use these available ECC methods to eradicate errors from extracted messages in digital images. In addition, this article did not cover the security needed in transferring hidden messages over the internet.

REFERENCES

[1] H. Yao, F. Mao, Z. Tang, and C. Qin, “High-fidelity dual-image reversible data hiding via prediction-error shift,” *Signal Processing*, vol. 170, p. 107447, 2020, DOI: 10.1016/j.sigpro.2019.107447.

[2] O. Evsutin, A. Melman, and R. Meshcheryakov, “Algorithm of error-free information embedding into the DCT domain of digital images based on the QIM method using adaptive masking of distortions,” *Signal Processing*, vol. 179, p. 107811, 2021, DOI: 10.1016/j.sigpro.2020.107811.

[3] B. Chen, G. W. Wornell, and S. Member, “Quantization Index Modulation: A Class of Provably Good Methods for Digital Watermarking and Information Embedding,” vol. 47, no. 4, pp. 1423–1443, 2001.

[4] Z. Jin, G. Feng, Y. Ren, and X. Zhang, "Feature extraction optimisation of JPEG steganalysis based on residual images R," *Signal Processing*, vol. 170, p. 107455, 2020, DOI: 10.1016/j.sigpro.2020.107455.

[5] H. Yao, H. Wei, C. Qin, and X. Zhang, "An improved first quantisation matrix estimation for nonaligned double compressed JPEG images," vol. 170, 2020, DOI: 10.1016/j.sigpro.2019.107430.

[6] M. Li *et al.*, “Extracting Spread-Spectrum Hidden Data From Digital Media,” vol. 8, no. 7, pp. 1201–1210, 2013.

[7] E. Porat, “The streaming k -mismatch problem,” pp. 1106–1125, 2019.

[8] P. Lugosch, "Learning Algorithms for Error Correction," no. April 2018.

[9] A. Dziech and J. Wassermann, “Application of Enhanced Hadamard Error Correcting Code in Video-Watermarking and his comparison to Reed-Solomon Code,” *MATEC Web Conf.*, vol. 125, pp. 1–8, 2017, DOI: 10.1051/matecconf/201712505007.

[10] M. Wootters, “(plus group testing!),” 2019.

[11] A. K. Ashutha K, “Error Minimization in BCH Codes,” *Int. J.*

Innov. Res. Electr. Electron. Instrum. Control Eng., vol. 4, no. 5, pp. 402–405, 2016, DOI: 10.17148/IJIREEICE.2016.4596.

[12] Q. Sun, D. He, Z. Zhang, and Q. Tian, "A secure and robust approach to scalable video authentication," *Proc. - IEEE Int. Conf. Multimed. Expo*, vol. 2, pp. 209–212, 2003, DOI: 10.1109/ICME.2003.1221590.

[13] A. Jiang, Y. Li, and J. Bruck, "Enhanced Error Correction via Language Processing," *NVM Work.*, 2015.

[14] O. Evsutin, A. Kokurina, R. Meshcheryakov, and O. Shumskaya, *The adaptive algorithm of information unmistakable embedding into digital images based on the discrete Fourier transformation*, vol. 77, no. 21. Multimedia Tools and Applications, 2018.

[15] S. Arunkumar, V. Subramaniyaswamy, V. Vijayakumar, N. Chilamkurti, and R. Logesh.

[16] Y. Yu, H. Yao, R. Ni, and Y. Zhao, "Detection of fake high definition for HEVC videos based on prediction mode feature," *Signal Processing*, vol. 166, p. 107269, 2020, DOI: 10.1016/j.sigpro.2019.107269.

[17] A. Sukumar, V. Subramaniyaswamy, V. Vijayakumar, and L. Ravi, "A secure multimedia steganography scheme using hybrid transform and support vector machine for cloud- based storage," *Multimed. Tools Appl.*, vol. 79, no. 15–16, pp. 10825–10849, 2020, DOI: 10.1007/s11042-019-08476-2.